



**Calhoun: The NPS Institutional Archive**  
**DSpace Repository**

---

Theses and Dissertations

1. Thesis and Dissertation Collection, all items

---

1979

# An in-depth study of the sensitivity of wave digital filters.

Sanaie-Fard, Masoud

Monterey, California. Naval Postgraduate School

---

<http://hdl.handle.net/10945/18758>

---

*Downloaded from NPS Archive: Calhoun*



Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

**Dudley Knox Library / Naval Postgraduate School**  
**411 Dyer Road / 1 University Circle**  
**Monterey, California USA 93943**

<http://www.nps.edu/library>

AN IN-DEPTH STUDY OF THE SENSITIVITY  
OF WAVE DIGITAL FILTERS

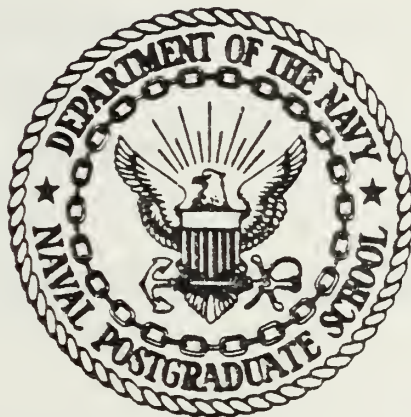
Masoud Sanaie-Fard

Thesis  
S1567



# NAVAL POSTGRADUATE SCHOOL

Monterey, California



## THESIS

AN IN-DEPTH STUDY OF THE SENSITIVITY  
OF WAVE DIGITAL FILTERS

by

Masoud Sanaie-Fard

June 1979

Thesis Advisor:

Sydney R. Parker

Approved for public release; distribution unlimited.

T183343





REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) An In-depth Study of the Sensitivity of Wave Digital Filters		5. TYPE OF REPORT & PERIOD COVERED Electrical Engineer; June 1979
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s)  Masoud Sanaie-Fard		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE June 1979
		13. NUMBER OF PAGES 239
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report)  Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report)  Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)  A detailed analytical and experimental study of the sensitivity of Wave Digital Filters has been conducted. The results indicate that the wave digital filter tends to achieve the same low sensitivity characteristic as the analogue circuit from which it was derived. Other results indicate relatively higher sensitivity to terminating resistance values compared with internal element values, lower sensitivity for algorithms derived from simple rather than multiple elements, and higher sensitivity at the critical		



frequencies. Finally the rms error due to the quantization in the number of bits in the multiplier coefficients has been measured at approximately 3 db per bit for the many examples tested.



Approved for public release; distribution unlimited.

An In-depth Study of the Sensitivity  
of Wave Digital Filters

by

Masoud Sanaie-Fard  
Commander, Iranian Navy  
BSEE, Electrical Engineering, Royal Naval  
Engineering College Manadon,  
Plymouth, England, 1967  
MSEE, Electrical Engineering, United States  
Naval Postgraduate School, 1978

Submitted in partial fulfillment of the  
requirements for the degree of

ELECTRICAL ENGINEER

from the  
NAVAL POSTGRADUATE SCHOOL  
June 1979



S1567  
c.1

## ABSTRACT

A detailed analytical and experimental study of the sensitivity of Wave Digital Filters has been conducted. The results indicate that the wave digital filter tends to achieve the same low sensitivity characteristic as the analogue circuit from which it was derived. Other results indicate relatively higher sensitivity to terminating resistance values compared with internal element values, lower sensitivity for algorithms derived from simple rather than multiple elements, and higher sensitivity at the critical frequencies. Finally the rms error due to the quantization in the number of bits in the multiplier coefficients has been measured at approximately 3 db per bit for the many examples tested.



## TABLE OF CONTENTS

I.	INTRODUCTION AND SCOPE - - - - -	17
A.	INTRODUCTION - - - - -	17
B.	SCOPE AND ORGANIZATION - - - - -	19
II.	GENERAL BACKGROUND - - - - -	22
A.	INTRODUCTION - - - - -	22
B.	GENERAL REVIEW OF DIGITAL FILTER DESIGN - - - - -	22
C.	METHODS OF DESIGNING THE DIGITAL FILTER - - - - -	23
1.	Impulse Invariance Method - - - - -	23
2.	Matched $z$ Transform - - - - -	24
3.	Designs Based on Numerical Solution of Analogue Differential Equations - - - - -	25
a.	Conventional Digital Filter Design - - - - -	25
b.	Wave Digital Filter Design - - - - -	29
D.	CHAIN OR A B C D MATRIX THEORY - - - - -	29
E.	CONCEPT OF DELAY FREE FEEDBACK PATH, OR DELAY FREE LOOP IN A TWO PORT NETWORK - - - - -	32
F.	SUMMARY - - - - -	35
III.	GENERAL DISCUSSION ON SENSITIVITY THEORY RELATED TO WAVE DIGITAL FILTERS - - - - -	36
A.	INTRODUCTION - - - - -	36
B.	DEFINITION OF SENSITIVITY - - - - -	36
C.	DEFINITION OF PARAMETER SENSITIVITY - - - - -	37
D.	QUANTIZATION ERROR IN DIGITAL FILTERS AND ITS SENSITIVITY EFFECT - - - - -	38
E.	SENSITIVITY OF LC LADDER STRUCTURES - - - - -	39
F.	PREVIOUS WORKS ON THE DEVELOPMENT OF THE WAVE DIGITAL FILTERS - - - - -	41



IV.	GENERAL THEORY OF WAVE DIGITAL FILTERS - - - - -	46
A.	INTRODUCTION - - - - -	46
B.	DERIVATION OF THE GENERALIZED TRANSFER RATIO FOR A TWO PORT WAVE DIGITAL FILTER ALGORITHM - - - - -	46
C.	REALIZATION OF TWO PORT WAVE DIGITAL STRUCTURE - - - - -	53
D.	DERIVATION OF GENERALIZED ALGORITHM WITH NO DELAY FREE PATH IN PORT ONE OR SIMILARLY WITH NO DELAY FREE PATH IN PORT TWO - - - - -	60
E.	TABULATION OF SIMPLE L AND C ELEMENTS IN SERIES AND SHUNT - - - - -	74
F.	FURTHER INVESTIGATION OF THE ALGORITHMS WITH NO DELAY FREE PATH IN PORT ONE, WITH THE ALGORITHMS WITH NO DELAY FREE PATH IN PORT TWO - - - - -	79
G.	DESIGN EXAMPLE - - - - -	80
H.	SUMMARY - - - - -	81
V.	SENSITIVITY ANALYSIS OF THE WAVE DIGITAL FILTERS DUE TO TRUNCATION IN THE NUMBER OF BITS - - - - -	82
A.	INTRODUCTION - - - - -	82
B.	COMPARATIVE STUDY OF THE ERROR DUE TO QUANTI- ZATION IN THE NUMBER OF BITS IN VARIOUS DIGITAL FILTER ALGORITHMS - - - - -	83
C.	STUDY OF THE RESULTS OBTAINED - - - - -	85
D.	COMPARISON OF RESULTS WITH EACH OTHER AND WITH THOSE PUBLISHED IN THE LITERATURE - - - - -	105
VI.	DERIVATION OF PARTIAL SENSITIVITY FUNCTIONS OF WAVE DIGITAL FILTERS - - - - -	112
A.	INTRODUCTION - - - - -	112
B.	DERIVATION OF THE SENSITIVITY FUNCTION OF WAVE DIGITAL FILTER DUE TO VARIATION IN MULTIPLIER COEFFICIENTS - - - - -	112
C.	DERIVATION OF SENSITIVITY FUNCTION OF WAVE DIGITAL FILTER IN THE FREQUENCY DOMAIN WITH RESPECT TO ITS ORIGINAL COMPONENT VALUES, I.E. L's, C's, $R_s$ , and $R_L$ - - - - -	118





D.	EXPERIMENTAL STUDY ON THE INTERNAL SENSITIVITY BEHAVIOR OF WAVE DIGITAL FILTERS - - - - -	-130
E.	ANALYSIS OF THE INTERNAL STRUCTURE OF THE WAVE DIGITAL FILTERS IN FREQUENCY DOMAIN USING FILTER PARTIAL SENSITIVITY FUNCTIONS - - - - -	-131
VII.	CONCLUSION - - - - -	-162
A.	INTRODUCTORY REMARKS - - - - -	-162
B.	SUMMARY OF THE IMPORTANT RESULTS IN THE FRE- QUENCY DOMAIN BEHAVIOR OF THE WAVE DIGITAL FILTER - - -	-162
C.	SUMMARY OF NEW THEORETICAL EXTENSIONS TO WAVE DIGITAL FILTER THEORY - - - - -	-163
D.	SUGGESTED FUTURE RESEARCH - - - - -	-164
APPENDIX 1	- - - - -	-172
A.	DESIGN OF LOW PASS LC FILTER WITH THE GIVEN SPECIFICATION - - - - -	-172
1.	Specification - - - - -	-172
2.	Data - - - - -	-172
3.	Design of Wave Digital Filter - - - - -	-172
4.	Design of the Conventional Digital Filter for Comparison Purposes - - - - -	-175
B.	COMPUTER PROGRAM NO. 1: Program for unity impulse response of the wave digital filter with no delay free path on port two with the given specification - - -	-184
C.	COMPUTER PROGRAM NO. 2: Program for transfer func- tion of the wave digital filter with no delay free path on port two with the given specification - - - - -	-188
D.	COMPUTER PROGRAM NO. 3: Program for unity impulse response of the conventional cascaded digital filter with the given specification - - - - -	-191
E.	COMPUTER PROGRAM NO. 4: Program for transfer func- tion of the conventional cascaded digital filter with the given specification - - - - -	-194
APPENDIX 2	- - - - -	-198
A.	COMPUTER PROGRAM NO. 5: Program to calculate the rms error due to truncation in the number of bits of wave digital filter multiplier coefficients - - - - -	-198



B.	COMPUTER PROGRAM NO. 6: Program to calculate the rms error due to truncation in the number of bits of complex wave digital filter multiplier coefficients - - - - -	204
C.	COMPUTER PROGRAM NO. 7: Program to calculate the rms error due to truncation in the number of bits of reduced parameter complex wave digital filter multiplier coefficients - - - - -	210
D.	COMPUTER PROGRAM NO. 8: Program to calculate the rms error due to truncation in the number of bits of the RLC components of the conventional direct digital filter - - - - -	215
E.	COMPUTER PROGRAM NO. 9: Program to calculate the rms error due to truncation in the number of bits of the RLC components of the conventional cascaded digital filter - - - - -	221
APPENDIX 3 - - - - -		227
A.	COMPUTER PROGRAM NO. 10: Program to calculate the sensitivity function of the seventh order low pass wave digital filter with respect to both wave digital filter multiplier coefficients and the original filter component values - - - - -	227
INITIAL DISTRIBUTION LIST - - - - -		239



# LIST OF TABLES

4.1a	Algorithms for simple L and C elements with no delay free path from $a_1$ to $b_1$ with sampling time T - - - -	75
4.1b	Iterative algorithms for simple L and C elements with no delay free path from $a_1$ to $b_1$ with sampling time T and with initial conditions i.e. $x_1, x_2, x_3, x_4$ set to zero - - - - -	76
4.2a	Algorithms for simple L and C elements with no delay free path from $a_2$ to $b_2$ with sampling time T - - - -	77
4.2b	Iterative algorithms for simple L and C elements with no delay free path from $a_2$ to $b_2$ with sampling time T and with initial conditions i.e. $x_1, x_2, x_3, x_4$ set to zero - - - - -	78
5.1	Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .5 db ripple Chebyshev filter with $R_s = 1.0$ (Case i) - - - - -	86
5.2	Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .1 db ripple Chebyshev filter with $R_s = 1.0$ (Case ii) - - - - -	87
5.3	Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass Butterworth filter with $R_s = 1.0$ (Case iii) - - - - -	88
5.4	Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .5 db ripple Chebyshev filter with $R_s = 0$ (Case iv) - - - - -	89
5.5	Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .1 db ripple Chebyshev filter with $R_s = 0$ (Case v) - - - - -	90
5.6	Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass Butterworth filter with $R_s = 0$ (Case vi) - - - - -	91





5.7	Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .5 db ripple Chebyshev filter with $R_s = 10$ (Case vii)	92
5.8	Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .1 db ripple Chebyshev filter with $R_s = 10$ (Case viii)	93
5.9	Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass Butterworth filter with $R_s = 10$ (Case ix)	94
A.1	Computer frequency response output for both wave digital filter and conventional cascaded digital filter	-181
A.2	Computer output for unity impulse response of both wave digital filter and conventional cascaded digital filter in the time domain	-182



## LIST OF FIGURES

2.1	Mapping of $f(s)$ into $f(z)$ as per relationship $z = e^{sT}$ - - - - -	26
2.2	Bilinear mapping of $f(s)$ into $f(z)$ as per relationship $s = 2(z-1)/T(z+1)$ - - - - -	26
2.3	Transformation of analogue filter $H(s)$ into digital filter using bilinear transform $S = 2(z-1)/T(z+1)$ - - - - -	28
2.4	A general two port structure with inputs $a_1$ and $a_2$ and outputs $b_1$ and $b_2$ , into and out of the system boundaries respectively - - - - -	30
2.5	A general two port structure with inputs $a_1$ and $a_2$ and outputs $b_1$ and $b_2$ , into and out of the system boundaries respectively - - - - -	30
4.1	Two generalized representations of a two port network. Fig. 4.1a representation in terms of voltage and current quantities. Fig. 4.1b representation in terms of scattering matrix wave quantities - - - - -	47
4.2	Wave flow diagram representing network $N_1$ of Fig. 4.1 - - - - -	51
4.3	General realization of a two port network $N_1$ with input and output waves - - - - -	54
4.4	A general two port "T" network with complex elements $z_1, z_2, z_3$ - - - - -	55
4.5	A general two port with series element only - - - - -	57
4.6	A general two port with shunt element only - - - - -	57
4.7	Signal flow graph of the direct delay free paths for three cascaded two ports with no delay free path from $a_1$ to $b_1$ , with multiplier coefficients $\sigma_1, \sigma_2$ , and $\sigma_3$ - - - - -	61
4.8	Signal flow graph of the direct delay free paths for a three cascaded two ports with no delay free path from $a_2$ to $b_2$ - - - - -	62
4.9	Signal flow graph of Fig. 4.7 modified to eliminate the existing delay free loop at the terminating load (no delay free path from $a_1$ to $b_1$ ) - - - - -	63



4.10	Signal flow graph of Fig. 4.8 modified to eliminate existing delay free loop at the terminating source (no delay free path from $a_2$ to $b_2$ ) - - - - -	64
4.11	Composite series and shunt element of example 4.2 - - - - -	69
5.1	Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the filter multiplier coefficients. Normalized seventh order low pass .5 db ripple Chebyshev filter with $R_s = 1.0$ , Case (i) - - - - -	95
5.2	Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the filter multiplier coefficients. Normalized seventh order low pass .1 db ripple Chebyshev filter with $R_s = 1.0$ , Case (ii) - - - - -	96
5.3	Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the filter multiplier coefficients. Normalized seventh order low pass Butterworth filter with $R_s = 1.0$ , Case (iii) - - - - -	97
5.4	Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the filter multiplier coefficients. Normalized seventh order low pass .5 db ripple Chebyshev filter with $R_s = 0$ , Case (iv) - - - - -	98
5.5	Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the filter multiplier coefficients. Normalized seventh order low pass .1 db ripple Chebyshev filter with $R_s = 0$ , Case (v) - - - - -	99
5.6	Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the filter multiplier coefficients. Normalized seventh order low pass Butterworth filter with $R_s = 0$ , Case (vi) - - - - -	100
5.7	Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the filter multiplier coefficients. Normalized seventh order low pass .5 db ripple Chebyshev filter with $R_s = 10$ , Case (vii) - - - - -	101





5.8	Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the filter multiplier coefficients. Normalized seventh order low pass .1 db ripple Chebyshev filter with $R_s = 10$ , Case (viii) - - - - -	102
5.9	Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the filter multiplier coefficients. Normalized seventh order low pass Butterworth filter with $R_s = 10$ , Case (ix) - - - - -	103
5.10	Internal resonances in a 7 <sup>th</sup> order doubly terminated LC filter - - - - -	106
5.11	Internal delay feedback in a wave digital filter after the 7 <sup>th</sup> order doubly terminated LC structure (with no delay free path on port two) - - - - -	106
5.12	Internal feedback in a complex wave digital filter patterned after the 7 <sup>th</sup> order doubly terminated LC structure (with no delay free path in port two) - - - - -	107
6.1	A general third order doubly terminated analogue low pass LC filter - - - - -	113
6.2	The wave digital filter derived from the analogue LC filter of Fig. 6.1 with no delay free path on port two - - - - -	113
6.3	Seventh order low pass analogue double resistively terminated LC filter - - - - -	120
6.4	The wave digital filter derived from the 7 <sup>th</sup> order analogue LC filter of Fig. 6.3 with no delay free path on port two - - - - -	120
6.5	The graphs of normalized sensitivity function $\partial H / \partial R_s \cdot R_s$ of various simple wave digital filters - - - - -	132
6.6	The graphs of normalized sensitivity functions $\partial H / \partial L_1 \cdot L_1$ and $\partial H / \partial \sigma_1 \cdot \sigma_1$ of various simple wave digital filters - - - - -	135
6.7	The graphs of normalized sensitivity function $\partial H / \partial C_2 \cdot C_2$ and $\partial H / \partial \sigma_2 \cdot \sigma_2$ of various simple wave digital filters - - - - -	138
6.8	The graphs of normalized sensitivity function $\partial H / \partial L_3 \cdot L_3$ and $\partial H / \partial \sigma_3 \cdot \sigma_3$ of various simple wave digital filters - - - - -	141



6.9	The graphs of normalized sensitivity function $\partial H / \partial C_4 \cdot C_4$ and $\partial H / \partial \sigma_4 \cdot \sigma_4$ of various simple wave digital filters - - - - -	144
6.10	The graphs of normalized sensitivity function $\partial H / \partial L_5 \cdot L_5$ and $\partial H / \partial \sigma_5 \cdot \sigma_5$ of various simple wave digital filters - - - - -	147
6.11	The graphs of normalized sensitivity function $\partial H / \partial C_6 \cdot C_6$ and $\partial H / \partial \sigma_6 \cdot \sigma_6$ of various simple wave digital filters - - - - -	150
6.12	The graphs of normalized sensitivity function $\partial H / \partial L_7 \cdot L_7$ and $\partial H / \partial \sigma_7 \cdot \sigma_7$ of various simple wave digital filters - - - - -	153
6.13	The graphs of normalized sensitivity function $\partial H / \partial R_L \cdot R_L$ and $\partial H / \partial \phi \cdot \phi$ of various simple wave digital filters - - - - -	156
7.1	Total combination of L and C in a two port network - - - - -	165
7.2	Separation of the complex two port networks shown in Fig. 7.1 into simple two or three port networks - - - - -	167
7.3	A three port signal graph with all possible combinations of delay free inputs and outputs - - - - -	168
7.4	Simple section wave digital filter signal flow graph of the complex series, shunt network of Fig. 7.1c - - - - -	168
7.5	A schematic diagram for the proposed adaptive wave digital filter - - - - -	170
A.1	Seventh order low pass analogue double resistively terminated LC filter - - - - -	173
A.2	Seventh order low pass .5 db ripple Chebyshev wave digital filter designed with no delay free path on port two after the seventh order low pass filter of Fig. A.1 - - - - -	173
A.3	Seventh order digitized and cascaded filter corresponding to seventh order analogue filter of Fig. A.1 - - - - -	179
A.4	Graph of the transfer function of the wave digital filter with the given specification in Section 1 - - - - -	183
A.5	Graph of the unity impulse response of wave digital filter with the given specification in Section 1 - - - - -	183



Dedicated to the  
Memory of My Father



## ACKNOWLEDGMENT

I would like to express gratefully and cordially my sincere gratitude to Professor Sydney R. Parker, for his constant guidance, encouragement, and generous devotion of the time, which made completion of this thesis possible. Working with him has been a great privilege and experience.

I am also very obliged to Professors Walter Ku, Cornell University, B. Peikari, Southern Methodist University, and R. E. Hamming for their generous help, and also the Iranian Navy for giving me this great opportunity.

Finally it would be inappropriate if I did not mention Ms. M. Marks who with great patience and sincerity did the typing and helped me in formatting this thesis.





## I. INTRODUCTION AND SCOPE

### A. INTRODUCTION

Signal processing is necessary in diverse areas of science and engineering, such as communication, social sciences, biomedical, control, radar, acoustics, telemetry, and intelligence and information gathering, etc. In general this process can be done with analogue (continuous time), or digital (discrete time, discrete amplitude) signals.

With the advent of LSI and VLSI, microprocessors, the need for efficient digital signal processing algorithms becomes more and more important. Much of the current literature is devoted to design of the linear algorithms, under the title of digital filters. Important factors in designing the digital algorithms are, time of the calculation, implementation, accuracy (error), etc.

Although there is evidence that direct digital filter design is possible [1], nearly all of the digital filter design algorithms use the analogue to digital transformation techniques. It is interesting to note that the rapid development of the digital signal processing is partly because of the existence of the well established theory on the analogue techniques and partly because of the abundance of the general purpose digital computers.

The digital computer and specially microprocessors, being physical objects from space and economical point of view, can only accommodate for finite precision in the size of the digital filter multiplier coefficients. Thus the need for digital filter algorithms with low sensitivity to digital filter multiplier coefficients arises.



Fettweis [2] in order to design a low sensitivity digital algorithm has proposed an alternative digital filter design method, namely wave digital filter, in which analogue LC circuit is transformed into discrete algorithm using wave or scattering matrix parameters. This approach in design is different from the conventional design techniques, because we have a new set of variables which are referred to as "incident and reflected wave parameters".

Wave digital filter design is believed to be difficult to understand, and is therefore generally avoided. But in reality this is not the case. In fact, to design a wave digital filter one need not go into the details of the algorithm development. One merely needs to know some basic facts and then can use the final wave digital filter equations and tables in order to design the required filter.

It is a well known fact that analogue LC networks have very low sensitivity to variation in LC component values. Upon this fact Fettweis and others have argued that, since the wave digital multiplier coefficients are derived from the LC parameters of the parent circuits, they should also have the same favorable low sensitivity characteristics to multiplier coefficients. Furthermore it is also known that the digital algorithms with low sensitivity to multiplier variations also exhibit minimum round-off noise due to quantization after multiplication of these multipliers with signals. As a result it is conjectured that the wave digital filter will have minimum round-off noise properties when compared with other digital filter algorithms. The purpose of this thesis is to analyze and check this conjecture.



## B. SCOPE AND ORGANIZATION

In order to develop and study the wave digital filter, the research in this thesis is divided into seven chapters. In Chapter II a brief discussion of the general digital filter theory is given. The presentation contains only selected and necessary background required for wave digital filter theory development in this thesis later on. Also included in this chapter are A B C D matrix theory, and the concept of the delay free feedback which plays an essential role in the wave digital filter theory development. Necessary sensitivity theory background used in the sensitivity analysis of digital filter is discussed briefly in Chapter III. The development of wave digital filter theory is done in Chapter IV. This development is straightforward and general in a sense that only one algorithm is developed for both series and shunt element. The effect of sampling interval is also introduced for the first time into the wave digital filter algorithms. Four useful tables of wave digital iterative algorithms for simple L and C elements in both series and shunt configuration are also given. The delay free path which plays an essential role in wave digital filter theory is emphasized throughout this chapter. The studies of the sensitivity of the wave digital filter to quantization in the number of bits of the multiplier coefficients is done in Chapter V. To do these studies in a fairly general sense, three different wave digital filter algorithms developed in Chapter IV are used with two different conventional digital filters for the given filter. A total of nine different filter types with different terminating source resistances were examined and compared with each other to arrive at a general conclusion. Chapter VI presents a specialized in-depth study of the sensitivity of the simple wave digital filter





algorithm. In this chapter sensitivity distribution along the filter structure was examined in order to understand the sensitivity behavior of the subsections of the wave digital filter on the overall sensitivity of the wave digital filter. Chapter VII summarizes the new results and proposes future research ideas related to the wave digital filter theory. There are three appendices. Appendix 1 includes an example to show the nearly exact equivalence between the wave digital filter and the conventional digital filter, both in the time domain and frequency domain when infinite precision arithmetic is used. The rest of the appendices include ten computer programs. The computer programs are used to derive the results of the main text. It is important to note that in order to facilitate the better understanding of the computer programs, explanatory remarks are made in the comment cards. These computer programs are in FORTRAN IV and can be used on any standard general purpose digital computer capable of dealing with FORTRAN IV scientific computer language.

Finally it is important to note that in this thesis for easy access, the references are given at the end of each chapter, rather than collectively at the end of the thesis.





## References

1. A. G. Constantinides, Digital Notch Filters, Electronics Lett. Vol. 5, No. 9, pp. 198-199, May 1969.
2. Alfred Fettweis, Digital filter structures related to classical filter networks, AEU, Vol. 25, No. 2, pp. 79-89, February 1971, [text is in English].



## II. GENERAL BACKGROUND

### A. INTRODUCTION

The main intent of this chapter is to briefly review general digital filter theory in order to establish the background necessary for the main subject of this thesis, i.e. wave digital filters. The design of the conventional digital filter is a well established subject. Thus it will be dealt very briefly. Also discussed in this chapter is the A B C D parameter matrix theory, which will be used later on followed by the concept of the delay-free feedback path or delay-free loop in the digital two port network which are used in deriving the causal wave digital filter algorithms in Chapter IV. z domain transform theory is assumed as a background and is not discussed.

### B. GENERAL REVIEW OF DIGITAL FILTER DESIGN

The design of electronic filters in the analogue domain is a well established subject, which not only includes very sophisticated techniques, but also has some very well established supporting computer programs as well. Much of the development of the digital filter has been directed towards the transfer of these results from the analogue domain to the digital domain.

In the most general sense a digital filter is a linear, shift invariant discrete time system which is realized using finite precision arithmetic. The design of the digital filters involves three basic steps:

- (1) the specification of the desired properties of the system



- (2) the approximation of these specifications using a causal discrete time system
- (3) the realization of the system using finite precision arithmetic

Note that these three steps are not independent of each other. In this thesis we are mainly interested in step 2 and to some extent on step 3.

### C. METHODS OF DESIGNING THE DIGITAL FILTER

An important class of techniques for designing infinite impulse response filters to be realized recursively is based on a transformation of a continuous time filter. This class comprises at least three techniques.

#### 1. Impulse Invariance Method

The impulse invariance method, also called standard  $z$  transformation (or standard  $z$ ), is a technique in which the impulse response of the derived digital filter is identical to the sampled impulse response of the continuous time filter. If the impulse response of the filter is  $h(t)$  then the sampled impulse response will be

$$h^*(t) = h(t) \cdot \delta_T(t) \quad (2.1)$$

where  $\delta_T(t)$  is the sampling function and is defined by

$$\delta_T(t) = \begin{cases} 1 & t=nT \\ 0 & t \neq nT \end{cases} \quad \text{where } n=0,1,2,\dots$$



It can be shown that the Laplace transform of  $h^*(t)$  will be

$$H^*(s) = \frac{1}{T} \sum_{K=-\infty}^{K=\infty} h(s + \frac{jK2\pi}{T}) \quad (2.2)$$

and the impulse invariance response of the filter will be

$$H(z) = H^*(s) \Big|_{z = e^{sT}} \quad (2.3)$$

from the relationship  $z = e^{sT}$  it is seen that the strips of width  $\frac{2\pi}{T}$  in the  $s$  plane map into the entire  $z$  plane as depicted in Figure 2.1, the left half of  $s$  plane strip maps into the interior of the unit circle, and the imaginary axis of the  $s$  plane maps onto the unit circle in such a way that each segment of length  $\frac{2\pi}{T}$  is mapped once around unit circle. Thus, the mapping is not a one to one mapping, and hence it can easily be shown that the impulse invariance response is only satisfactory if  $H(s)$  is band limited. And as in most cases if  $H(s)$  is not sufficiently band limited  $H(z)$  is an aliased version of  $H(s)$ . Therefore the technique is used for narrow band filter design or else  $H(s)$  is broken into cascaded subsections of first order and second order sections with guard filters in between, which in some cases is a tedious job. Also it is clear from equation (2.2) that due to the  $\frac{1}{T}$  multiplier, the digital filters derived using the impulse invariance method have a gain approximately  $\frac{1}{T}$  to that of continuous time filter, which should be taken into account in the design.

## 2. Matched z Transform

This procedure is based on mapping the poles and zeros of the continuous time filter by the substitution of  $(s-s_1) \rightarrow (1-e^{s_1 T}, z^{-1})$ .





This means that poles of  $H(z)$  will be identical to those obtained by impulse invariant method; however the zeros will not correspond.

### 5. Designs Based on Numerical Solution of Analogue Differential Equations

#### a. Conventional Digital Filter Design

In this method, the differential equation of the analogue filter is approximated by a recursive equation, which is the standard procedure in Numerical Analysis. There are three basic numerical integration techniques, namely,

- (i) Euler forward integration
- (ii) Euler backward integration
- (iii) bilinear integration

All these techniques plus many others are described fully in the literature. And there is no need to go into details for all of these techniques, but because of the importance of bilinear transformation we describe it briefly.

The approach uses the algebraic transformation

$$s = \frac{2}{T} \left[ \frac{1-z^{-1}}{1+z^{-1}} \right] \quad (2.4)$$

to derive the system transfer function of the digital filter such that

$$H(z) = H(s) \bigg|_{s = \frac{2}{T} \left( \frac{1-z^{-1}}{1+z^{-1}} \right)} \quad (2.5)$$

This transformation has the effect of mapping the entire left half  $s$  plane into the inside of the unit circle and entire right half of the  $s$  plane into the outside of the unit circle as shown in Figure 2.2. This results in a nonlinear warping of the frequency scale according to the relation



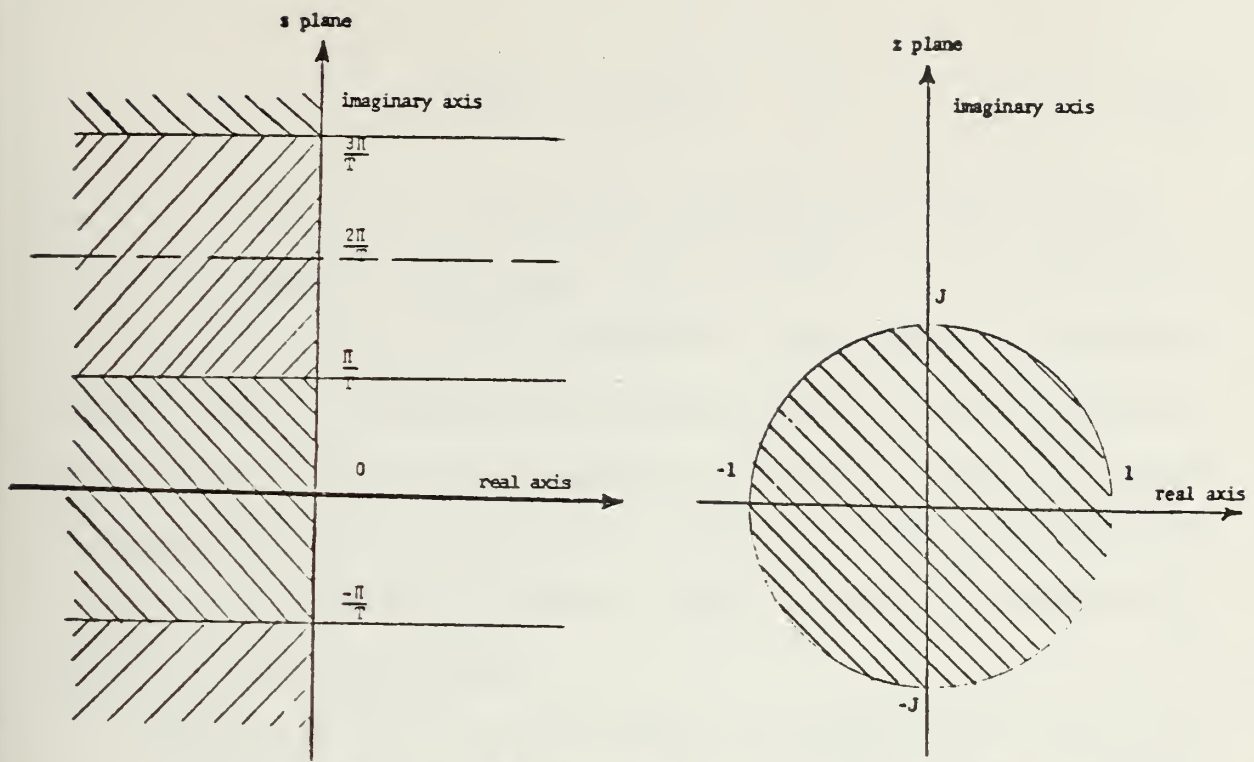


Fig. 2.1. Mapping of  $f(s)$  into  $f(z)$  as per relationship  $z = e^{sT}$ . Note that the area between the strips  $\pi/T$  and  $-\pi/T$  map into the entire  $z$  plane in such a way that the area on the left half  $s$  plane strip maps into the inside of the unit circle and the area on the right half  $s$  plane strip maps outside the unit circle. The process is repeated infinitely many times thus mapping or transformation is not one to one.

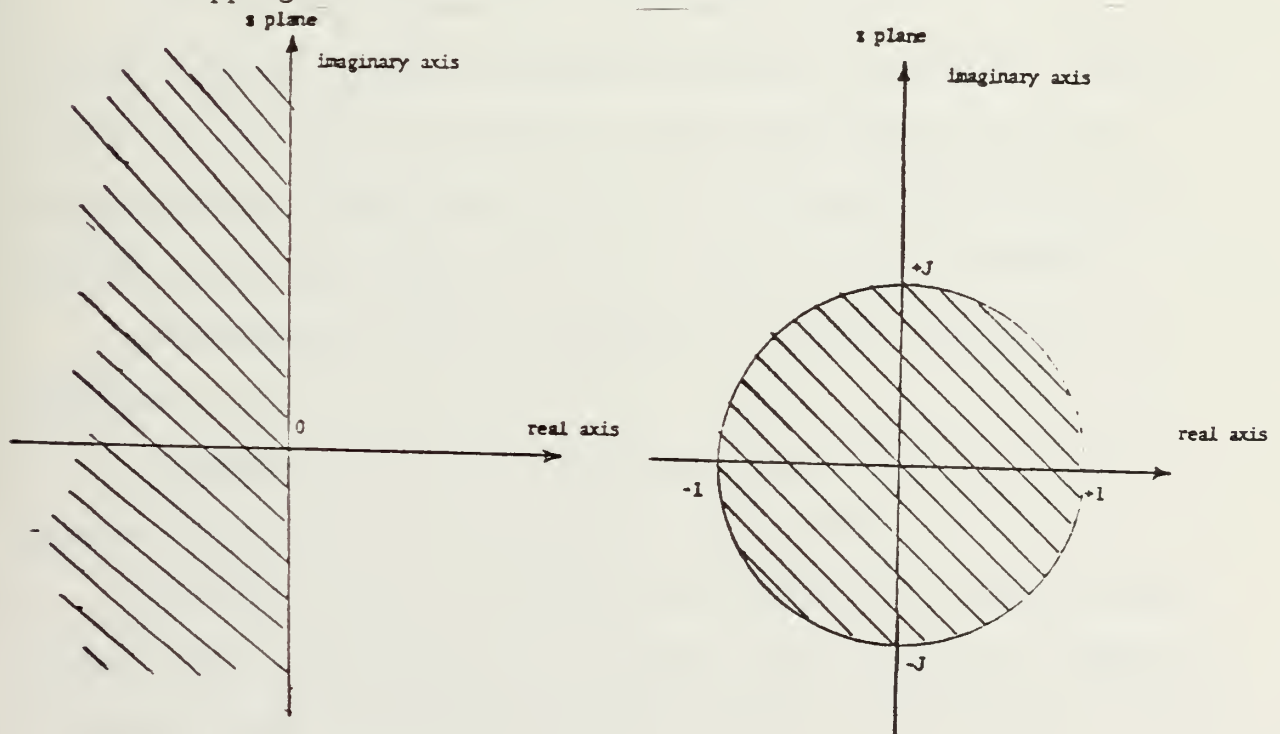


Fig. 2.2. Bilinear mapping of  $f(s)$  into  $f(z)$  as per relationship  $s = 2(z-1)/(T(z+1))$ . Note that the transformation is a one to one mapping.



$$\frac{\omega_c T}{2} = \tan \frac{\Omega_{cd} T}{2} \quad (2.6)$$

where  $\omega_c$  is the critical frequency of analogue filter and  $\Omega_{cd}$  is the critical frequency of the digital filter. Because of this warping of the frequency scale this design technique is most useful in obtaining digital design of filters whose frequency response can be divided up into a finite number of pass bands and stop bands in which the response is essentially constant. Figure 2.3 shows the frequency response of an analogue filter and its approximated digital frequency response using bilinear transform techniques.

From Figure 2.3 it is obvious that if we require the critical frequency of the digital filter to be say at  $\Omega_{cd}$  then we have to frequency scale the critical frequency of the analogue filter by a factor

$$\text{factor} = \frac{T}{2 \tan \frac{\Omega_{cd} T}{2}} \omega_c . \quad (2.7)$$

Typical frequency selective analogue filters are Butterworth, Chebyshev, and Elliptic filters. Note that all these filters have closed form analogue design formulas and by the use of bilinear transformation we can easily get approximate closed form digital filter algorithms.

A Butterworth filter is a monotonic in the pass band and in the stop band.

A Chebyshev filter has an equiripple characteristic in the pass band and monotonic in the stop band, or vice versa.

An Elliptic filter is equiripple in both pass band and stop band. Clearly these properties will be preserved when the filter is mapped to a digital filter with the approximated bilinear transformation as shown in Figure 2.3.



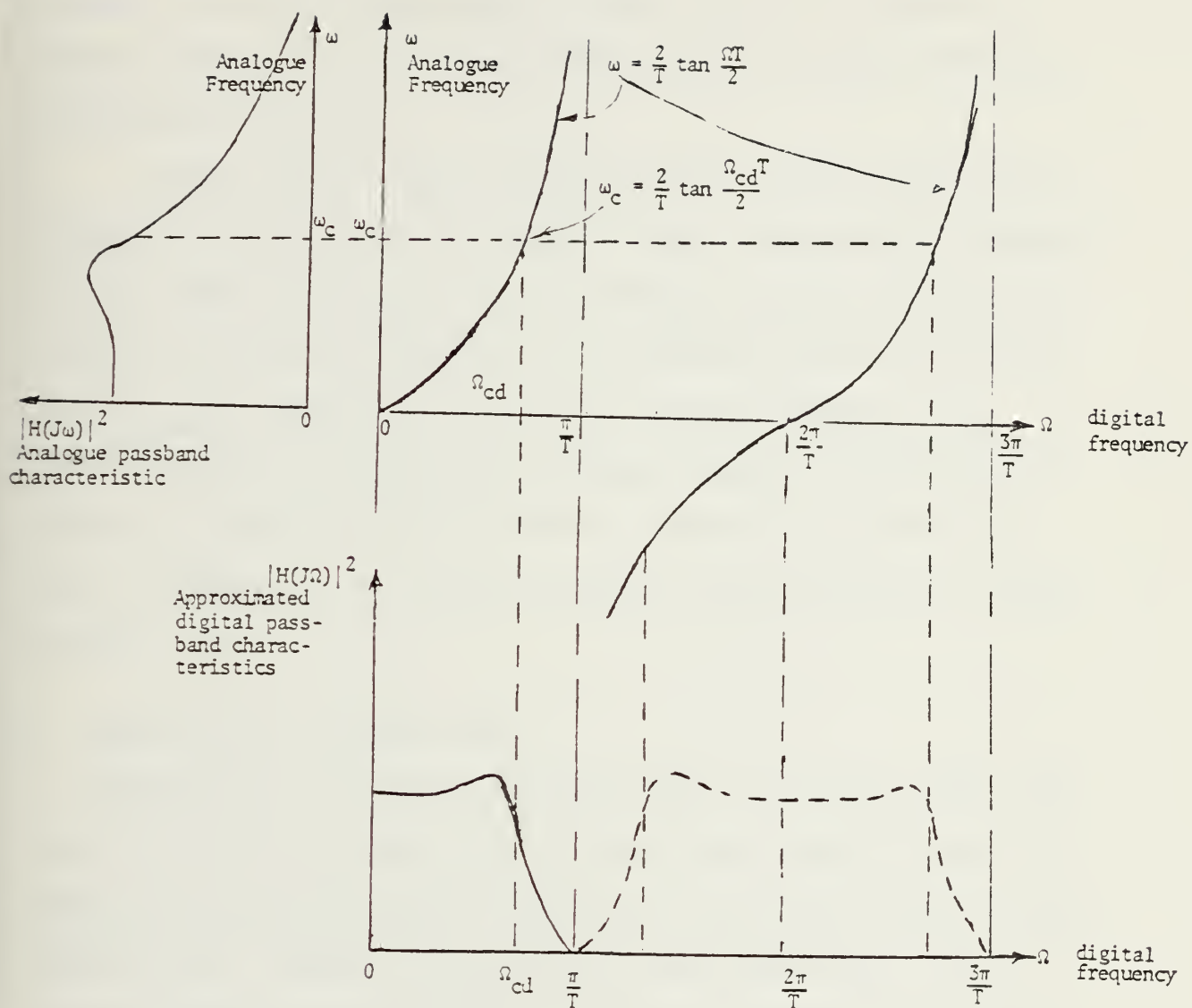


Figure 2.3. Transformation of analogue filter  $H(s)$  into digital filter using bilinear transform  $S = 2(z-1)/T(z+1)$ .





## b. Wave Digital Filter Design

This technique basically uses the bilinear transformation of the analogue to digital design exactly the same as part a iii, but the attempt is made only on LC filters having input and output terminating resistances of  $R_1$  and  $R_2$ .

In this technique each reactive element in the analogue ladder structure is transformed into a two port signal flow structure using the bilinear transformation and wave flow techniques, and at the same time matching the port one impedance of the derived two port structure to the port two impedance of the previously derived element, thus eliminating mismatch between the succeeding elements. The details of this and its implementation are left for Chapter IV. Note that the idea is a very basic one and can be applied on many varieties of common filters.

## D. CHAIN OR A B C D MATRIX THEORY

The analysis of any passive linear network with several inputs and outputs can be done in many ways. The most usual ones are signal flow graph, system matrix equations, input/output algorithms, transform matrix equations, etc. However, for systems of order higher than 2, most of the above analysis becomes tedious and prone to mistakes due to system complexity. A most useful and convenient method of dealing with a complex system is, whenever possible, to break the system into several subsystems and interrelate these subsystems by a chain matrix, thus allowing each subsystem to be analyzed and investigated separately one at a time, without even thinking about the rest of the system. To illustrate the point consider the network N of Figure 2.4 with wave inputs  $a_1$  and  $a_2$  and wave outputs  $b_1$  and  $b_2$ . The relationship between port 2 and port 1 for this network can be written as



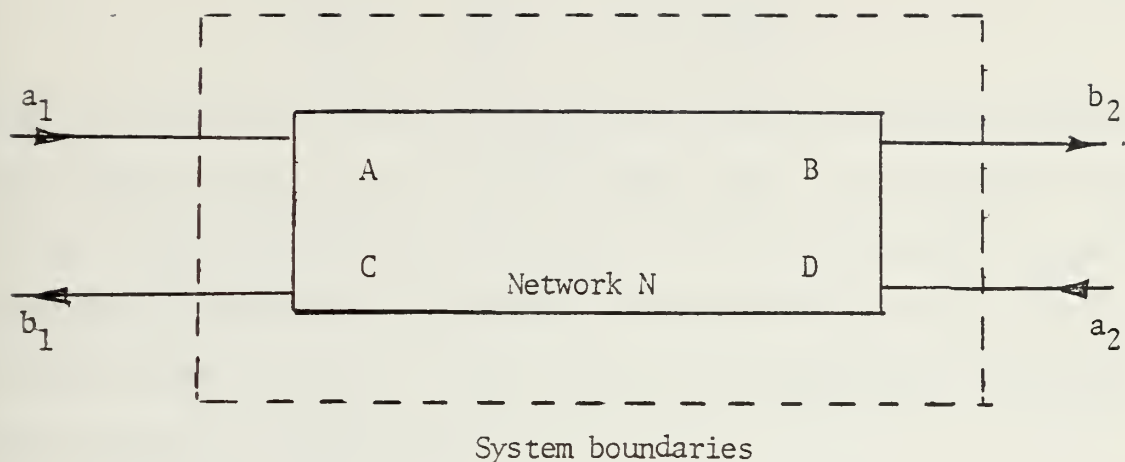


Fig. 2.4. A general two port structure with inputs  $a_1$  and  $a_2$  and outputs  $b_1$  and  $b_2$ , into and out of the system boundaries respectively.

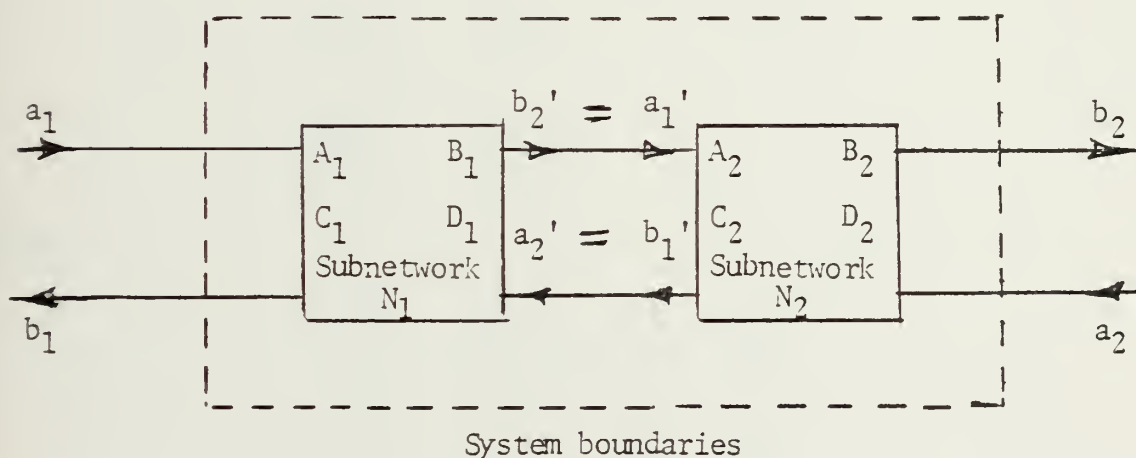


Fig. 2.5. A general two port structure with inputs  $a_1$  and  $a_2$  and outputs  $b_1$  and  $b_2$  into and out of the system boundaries respectively. Note that the system of Fig. 2.4 is equivalent to the system of Fig. 2.5 if the inputs ( $a_1$  and  $a_2$ ) and outputs ( $b_1$  and  $b_2$ ) are exactly the same.



$$\begin{bmatrix} b_2 \\ a_2 \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \quad (2.8)$$

Now if we can break network N into several subnetworks inside the dotted line without touching the input and output ports, the resulting network will be exactly the same as network N. Note that the networks  $N_1$  and  $N_2$  resulting from network N do not necessarily have to have equal subsections as shown in Fig. 2.5. Note also how the consistency in the wave-flow direction is maintained.

Thus from Fig. 2.4

$$\begin{bmatrix} b_2' \\ a_2' \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \quad (2.9)$$

$$\begin{bmatrix} b_2' \\ a_2' \end{bmatrix} = \begin{bmatrix} a_1' \\ b_1' \end{bmatrix} \quad (2.10)$$

and

$$\begin{bmatrix} b_2 \\ a_2 \end{bmatrix} = \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} a_1' \\ b_1' \end{bmatrix} \quad (2.11)$$

$$\begin{bmatrix} b_2 \\ a_2 \end{bmatrix} = \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \quad (2.12)$$

or

$$\begin{bmatrix} b_2 \\ a_2 \end{bmatrix} = \begin{bmatrix} A_1 A_2 + B_2 C_1 & A_2 B_1 + B_2 D_1 \\ A_1 C_2 + C_1 D_2 & C_2 B_1 + D_1 D_2 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \quad (2.13)$$



Note that equations (2.8), (2.13) are identical. This simple example clearly demonstrates how a system when made up of only two simple subsystems  $N_1$  and  $N_2$  which are easy to be analyzed each separately, when combined into the system  $N$  becomes a complex system, and very difficult to get analyzed.

Note that in the first case it is a very easy job to analyze subelements  $A_1$ ,  $B_1$ ,  $C_1$  and  $D_1$  of system  $N_1$  or that of the system  $N_2$ . While clearly the analysis of the element  $A = A_1A_2 + B_2C_1$  etc. of the system  $N$  will not be an easy job, and in most cases when the system is made up of more than 2 subsystems it is a tedious if not an impossible task. Thus equation (2.12) clearly demonstrates the fundamental property of the chain matrix analysis.

Whenever two or more than two pairs are connected in cascade the chain matrix of the composite network will be the product of the individual chain matrices. Since in general matrix multiplication is not commutative, it is important that the matrices be multiplied in the same order as the circuits are cascaded. It can easily be shown that if all the individual matrices are reciprocal the composite two port will also be reciprocal.

#### E. CONCEPT OF DELAY FREE FEEDBACK PATH, OR DELAY FREE LOOP IN A TWO PORT NETWORK

As mentioned in section II-D the relationship between port two and port one of a subsection of a general causal system such as that of Fig. 2.5 can best be described by the equation (2.8), i.e.:

$$\begin{bmatrix} b_2' \\ a_2' \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} \quad (2.14)$$

where  $b_1$ ,  $a_1$  are the output and input quantities at port one and  $b_2'$ ,  $a_2'$  are the output and input quantities at port two. This equation can





be rewritten in terms of input quantities in the following form

$$\begin{bmatrix} b_1 \\ b_2' \end{bmatrix} = \begin{bmatrix} \frac{C_1}{D_1} & \frac{1}{D_1} \\ \frac{A_1 D_1 - B_1 C_1}{D_1} & \frac{B_1}{D_1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2' \end{bmatrix} \quad (2.15)$$

or we can write equation (2.15) in the simpler form of

$$\begin{bmatrix} b_1 \\ b_2' \end{bmatrix} = \begin{bmatrix} f_1(z) & f_2(z) \\ f_3(z) & f_4(z) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2' \end{bmatrix} \quad (2.16)$$

Note that for a causal digital system the values  $f_1(z)$ ,  $f_2(z)$ ,  $f_3(z)$  and  $f_4(z)$  are of the form

$$f(z) = \frac{a_0 + a_1 z^{-1} + \dots + a_i z^{-i} + \dots + a_n z^{-n}}{1 + b_1 z^{-1} + \dots + b_i z^{-i} + \dots + b_n z^{-n}}. \quad (2.17)$$

Note also that when written in terms of positive exponents of  $z$ , the order of  $z$  in the numerator must be equal to or less than the order of the denominator for causality. With this in mind the iterative equations derived from equation (2.15) are

$$b_1(n) = \alpha_1 a_1(n) + \beta_1 a_2'(n) + \text{weighted values of past inputs at port one and port two plus weighted values of past outputs at port one} \quad (2.18)$$

and

$$b_2'(n) = \alpha_2 a_1(n) + \beta_2 a_2'(n) + \text{weighted values of past inputs at port one and port two plus weighted values of past outputs at port two} \quad (2.19)$$

Note that  $\alpha_1$ ,  $\alpha_2$ ,  $\beta_1$ ,  $\beta_2$  are merely weighting constants, and equations (2.18) and (2.19) are both causal and realizable.



Now if the input  $a_2'(n)$  at the port two is a function of  $b_2'(n)$  of the port two, i.e.

$$a_2'(n) = Kb_2'(n) + \text{weighted values of past input and output values at port two} \quad (2.20)$$

which is the case for most cascaded two port systems such as the one shown in Fig. 2.5 the equations (2.18) and (2.19) further reduce to

$$b_1(n) = \alpha_1 a_1(n) + \beta_1 Kb_2'(n) + \text{weighted values of past inputs and outputs at port one and port two} \quad (2.21)$$

and

$$b_2'(n) = \alpha_2 a_1(n) + \beta_2 Kb_2'(n) + \text{weighted values of past inputs at port one and port two plus the weighted values of past outputs at port two} \quad (2.22)$$

Note that the iterative equation (2.22) is not causal since for the calculation of  $b_2'(n)$  it requires  $b_2'(n)$  which is not possible. Thus for the causality either  $K$  must be equal to zero or  $\beta_2$  must be equal to zero which are two distinct cases.

#### Case 1

By considering Fig. 2.5 it can be seen that  $a_2'$ ,  $b_2'$  are merely the port one quantities of the second stage. Thus if we are going to consider the first stage only we cannot force  $K$  to be equal to zero. Thus the only variable left is  $\beta_2$  and by making  $\beta_2$  equal to zero we can make equation (2.22) a valid equation. This condition is referred to the case of no internal delay free path from  $a_2'$  to  $b_2'$  or no delay free path in port two. Thus to implement this condition the order of the numerator of the function  $f_4(z) = \frac{B_1(z)}{D_1(z)}$  must be at least one order lower than the denominator.



## Case 2

As it was noted in the Case 1 since  $a_2'$ ,  $b_2'$  are merely port one quantities of the second stage. If we can make  $b_1'(n)$  of the second stage to be independent of the  $a_1'(n)$  of the second stage and only dependent on the past inputs and outputs of the port one of the second stage, then we have actually managed to make  $a_2'(n)$  of the first stage to be independent of  $b_2'(n)$  of the first stage. This can be done by forcing  $\alpha_1 = 0$  in the second stage. This condition is referred to the case of no internal delay free path from  $a_1$  to  $b_1$  or no delay free path in port one. Note that to implement this condition the order of the numerator of the function  $f_3(z) = \frac{A_1(z)D_1(z) - B_1(z)C_1(z)}{D_1(z)}$  must be at least one order lower than that of the denominator.

## F. SUMMARY

In this chapter we have reviewed briefly the ground work required for the matched two port wave digital filter theory and design.

The contents of this chapter are used throughout this thesis. No particular mention of any reference has been made since most of the subjects discussed are well established and details can be found in most digital filter design handbooks or papers. It is worthwhile to note that the concept of delay free feedback path, though important, in most papers reviewed were merely stated without any proof. Thus in this chapter an attempt was made to prove it in the most general sense.



### III. GENERAL DISCUSSION ON SENSITIVITY THEORY RELATED TO WAVE DIGITAL FILTERS

#### A. INTRODUCTION

The main intent of this chapter is to start with the low sensitivity properties of the doubly terminated analogue LC structure and then extend this property to the wave digital filter. Later on in the chapter we explain briefly the development of wave digital filter theory. It is not the intention to go into details, but merely to give an overview of previous works for which full development is available in the references.

Another objective of this chapter is to briefly discuss the different wave digital filter structures and algorithms which are all called wave digital filters, each structure having its own characteristics and limitations. The newcomer to this field will be astonished by the many different structures and algorithms which are present in the literature; all of them are offered under the same name of wave digital filter.

Later in this chapter a natural development of the wave digital filter theory is traced from the initial conjecture of Fettweis up to the present day state of the art.

#### B. DEFINITION OF SENSITIVITY

The term sensitivity of a certain filter structure has a broad meaning, and the literature is full of different definitions for sensitivity functions meeting a specified requirement. But in general all of the definitions end up more or less to the following:





As the filter element values or coefficients are varied about their true value, the filter pole and zero locations are shifted correspondingly, causing a change in the magnitude and phase characteristic of the filter. A mean shift in the characteristic can be computed on a point by point basis and used as a figure of merit, but it may be meaningless when determining whether or not the original filter specifications are met. Therefore sensitivity must be interpreted in terms of several factors to include such parameters as bandwidth, cut-off frequency, ripple in the pass and stop band, etc. Several sensitivity functions are most commonly used. They are

- (i) Q sensitivity and pole frequency sensitivity
- (ii) Root sensitivity
- (iii) Coefficient sensitivity
- (iv) Frequency response sensitivity, etc.

Note that most of these sensitivity functions are meaningful and for most cases there is a relationship between most of these sensitivity functions with each other [12].

Since in this thesis we are mainly interested in the parameter or coefficient sensitivity, thus no discussion of the other sensitivity functions will be made.

### C. DEFINITION OF PARAMETER SENSITIVITY

For the function  $H(C_1, C_2, \dots, C_i, \dots, C_n)$ , where  $H$  is a function of multiparameter  $(C_1, C_2, \dots, C_i, \dots, C_n)$ , the parameter sensitivity  $H$  of the system to any change of variables or elements  $C_i$  is defined in several ways

$$i) \quad S = \frac{\partial H}{\partial C_i} \cdot \frac{C_i}{H}, \quad \begin{array}{l} \text{logarithmic or} \\ \text{factorial sensitivity} \end{array} \quad (3.1)$$



$$\text{ii) } S = \frac{\partial H}{\partial C_i}, \text{ derivative sensitivity} \quad (3.2)$$

$$\text{iii) } S = \frac{\partial H}{\partial C_i} \cdot \frac{1}{H} \text{ or } S = \frac{\partial H}{\partial C_i} C_i, \text{ semi-logarithmic sensitivity}$$

and all of these sensitivity functions are discussed fully in the literature [12], [13], [14], [15].

#### D. QUANTIZATION ERROR IN DIGITAL FILTERS AND ITS SENSITIVITY EFFECT

Although tolerance problems which exist in the physical world in the case of analogue filters, do not exist as such for digital filters, still there is an interest in obtaining structures with low sensitivity to parameter variations. There are several reasons for this. The primary reason is the fact that the structures with low sensitivity are less affected by coefficient truncation. Thus element values or coefficients with shorter word length are sufficient for meeting a given specification. The second reason stated and proved by Fettweis [10] is that there exists a relationship between sensitivity with respect to multiplier variation and the round off noise at the output of a digital filter. Thus any improvement in the sensitivity would result in a reduction in the corresponding round-off noise at the output.

In order to compare the coefficient sensitivity of different digital structures Ku and Ng [8] have used a root mean square error criteria in the frequency response given by

$$E_{\text{rms}} = \left[ \frac{1}{N+1} \sum_{i=0}^N [W(\omega_i) [\Delta H(\omega_i)]^2] \right]^{\frac{1}{2}} \quad (3.3)$$

Note that this criteria is in the frequency domain and is based on the deviation of the magnitude of the finite precision output from the ideal or infinite precision output.



Thus  $\Delta H(\omega_i)$  is defined as

$$\Delta H(\omega_i) = |H(\omega_i) - H_0(\omega_i)| \quad (3.4)$$

where  $H_0(\omega_i)$  is the magnitude of the ideal output at frequency  $\omega_i$ , and  $H(\omega_i)$  is the magnitude of the finite precision output at frequency  $\omega_i$ , and  $W(\omega_i)$  is the weight chosen to reflect the relative importance of error at various frequencies, and  $N$  is the number of sample points in the frequency domain.

Furthermore in order to equalize the effect of the various coefficients, floating point arithmetic is used rather than fixed point arithmetic, during the process of rounding off of the coefficient to the required number of the bits. Note that more details and implementation of the idea are left for Chapter V where we investigate case studies based on bit quantization error.

#### E. SENSITIVITY OF LC LADDER STRUCTURES

It is a well known fact that the doubly terminated analogue LC ladder structures are relatively insensitive to element value changes. To prove this fact most researchers have used the theory of the total conservation of input and output power quantities in a reactive lossless network, the discussion of which is interesting but it not in the scope of this thesis. A very good review is made in this respect by Renner and Gupta [7].

Thus, it is a reasonable assumption that the digital filter derived directly from the topology of a resistively terminated analogue lossless structure would have the same favorable sensitivity properties as its analogue counterpart.

To show the low sensitivity of wave digital filters by the conventional  $n$  port scattering matrix network theory, Fettweis [10] defines the



instantaneous "pseudopower" transmitted through a wave port k with port input resistance  $R_k$  by the means of the equation

$$p_k = \left( \frac{a_k^2 - b_k^2}{R_k} \right) \quad (3-5)$$

where  $p_k$  is the total instantaneous pseudopower input through port k and  $a_k$  is the total input wave at port k and  $b_k$  is the total reflected output wave at port k. Similarly he defines steady state pseudopower by the relationship

$$P_k = \left( \frac{|A_k|^2 - |B_k|^2}{R_k} \right) \quad (3-6)$$

where  $P_k$  and  $A_k$  and  $B_k$  are the steady state values of  $p_k$ ,  $a_k$ ,  $b_k$ , respectively at an arbitrary frequency "f". In this way he then shows that for all wave digital building blocks which are derived from the LC doubly terminated ladder structures, the total sum of instantaneous pseudopower absorbed by all ports is equal to zero, i.e.

$$\sum_{k=1}^n p_k = 0$$

where n is the total number of the ports of the wave digital subelement or building block and  $p_k$  is the instantaneous pseudopower at port k.

Similarly he also proves that the total steady state pseudopower absorbed by all building blocks of a terminated wave digital filter at an arbitrary complex frequency "f" is equal to zero, i.e.

$$\sum_{k=1}^n P_k = 0$$





where  $P_k$  is the steady state pseudopower at an arbitrary complex frequency  $f$ , and  $n$  is the total number of the ports of the wave digital building block. The low sensitivity property then follows from the zero pseudopower relationships. Although power has physical meaning in analogue circuits its meaning in digital filter algorithms is nebulous.

#### F. PREVIOUS WORKS ON THE DEVELOPMENT OF THE WAVE DIGITAL FILTERS

In the previous section we discussed the favorable low sensitivity to parameter variation properties of LC ladder structures. It was this fact that led Fettweis to propose the idea of wave digital filters in 1970 [3], derived from the doubly resistive terminated LC ladder structures. Since then numerous papers relating to the wave digital filters have been presented.

To understand the wave digital filters and the state of art, it is necessary to summarize the state of evolution of the wave digital filters.

Partial credit of the development of the wave digital filters can also be given to Richards [1], Kuroda, Ozaki and Ishii [2] and numerous other researchers who contributed to the development and synthesis of the strip line filters and also resistor-transmission line circuits.

The basic idea behind wave digital filter development from the beginning was to keep the desired low sensitivity of lumped LC resistively terminated filters in the already developed analogue domain and transfer it to digital domain with very minor modifications.

Ideal strip line filter and transmission line circuits having inherent LC structure is an ideal starting point. To avoid the loading effect of one element of the LC strip line upon the succeeding and preceding elements, Ozaki and Ishii [2] introduced the idea of inserting unity element, or the lossless transmission line acting as an ideal transformer (which was already developed by Kuroda and known as Kuroda's identity), between the stages as



a lossless matching element.

In a more detailed paper Fettweis [4], in order to arrive at realizable signal flow diagram, used wave quantities instead of the usual current and voltage input quantities to the conventional filter. Actually in using the wave quantities as input and output signal, Fettweis made use of the already developed scattering parameter matrix theory of the  $n$  port networks. In order to avoid the loading effect of one section upon the other, and thus to eliminate the unwanted mismatch reflections between sections, he employed unit element or impedance transformer, known as Kuroda's identity [2].

Although with the introduction of the unity element a desirable result was achieved in realizing the LC filter structures, this introduction further increased the number of required operations (mainly multiplications) over the conventional digital filter.

In order to avoid the unity element several attempts were made by different researchers, Sedlmeyer and Fettweis [5], Van Haften and Chirlain [6], and others, to eliminate the need for unity elements or impedance transformers between the cascading sections. One of the attempts was made by Fettweis [5] and later on the method was renamed by Ku and Ng [8] as the "new Fettweis method." The new Fettweis method forced one of the scattering multiplier coefficients in the three port network to be equal to 1, thus a two port element was made from the three port element. By starting at one port and progressing forward towards the other port, and by suitable choice of input impedance of the next section (i.e. matching the output impedance of one section to the succeeding section), the need for unity element or impedance transformer matching was eliminated.



Van Haften and Chirlain [6] next came up with the idea of cascading the wave digital three port section directly without the unity element but taking into account the approximate attenuation of one section upon the next section, thus eliminating the need for unity element and extra multiplications associated with it. At the same time this introduced errors associated with the calculation of the approximate attenuation of one section upon the other.

S. Erfani and B. Peikari [9] and N. S. Swamy and K. S. Thyagarajan [11] at the same time came up with a new approach to eliminate the unity element. This technique is the basis of the further investigation in this thesis, and details are given in Chapter IV and other chapters.



## References

1. Paul I. Richards, Resistor transmission line circuits, Proceedings of IRE, February 1948, pages 217-220.
2. H. Ozaki and J. Ishii, Synthesis of a class of strip line filters, IRE Transactions on Circuit Theory, Vol. CT-5, June 1958, pages 104-109.
3. Alfred Fettweis, Some principles of designing digital filters imitating classical filter structures, IEE Transactions on Circuit Theory, March 1971, pages 314-316.
4. Alfred Fettweis, Digital filter structures related to classical filter networks, Archiv fur Elektronik und Vbertragungstechnik, Vol. 25, No. 2, February 1971, [text written in English].
5. Axel Sedlmeyer and Alfred Fettweis, Digital filters with true ladder configuration, IEE, Circuit Theory and Applications, Vol. 1, 5-10, 1973.
6. D. Van Haften, An analysis of errors in wave digital filters, Doctoral dissertation, 1977, Stevens University.
7. K. Renner and S. C. Gupta, On the design of wave digital filters with low sensitivity properties, IEEE Transactions on Circuit Theory, Vol. CT-20, No. 5, September 1973, pages 555-567.
8. W. H. Ku and S. M. Ng, Floating point coefficient sensitivity and round-off noise of recursive digital filters realized in ladder structures, IEE Transactions of Circuits and Systems, Vol. CAS-22, No. 12, December 1975.
9. S. Erfani, Design of fixed and variable digital filters using generalized delay units, Ph.D. dissertation, 1976, Southern Methodist University, Dallas, Texas.
10. Alfred Fettweis, Pseudopassivity, sensitivity, and stability of wave digital filters, IEEE Transactions on Circuit Theory, Vol. CT-19, No. 6, November 1972.
11. K. S. Thyagarajan, One and two dimensional wave digital filters with low coefficient sensitivities, 1977, Doctoral dissertation, Concordia University, Montreal, Canada.
12. G. C. Temes and S. K. Mitra, Modern filter theory and design, Chapter 8, 1973, Published by John Wiley and Sons.
13. S. R. Parker, Sensitivity analysis and models of nonlinear circuits, IEEE Transactions on Circuit Theory, Vol. CT-16, No. 4, November 1969.





14. R. Tomovic and M. Vukobratovic, General sensitivity theory, 1972, American Elsevier Publishing Company, Inc., New York.
15. S. R. Parker, Sensitivity old questions, some new answers, IEE Transactions on Circuit Theory, Vol. CT-18, No. 1, January 1971.



#### IV. GENERAL THEORY OF THE WAVE DIGITAL FILTER

##### A. INTRODUCTION

The intent of this chapter is to derive the generalized algorithm for a two port wave digital filter on a step by step basis using the principles of the circuit theory, matrix algebra, and scattering matrix wave theory.

The resulting algorithms are summarized in two tables in section E. Also, two illustrative examples are given, example one being a simple element, suitable for matched source; and example two being a complex element, suitable for a matched load.

It must be emphasized that the techniques used to derive the wave digital algorithms in this chapter are more general than the previous works, in the sense that only one algorithm is derived for both series and shunt element. Also the effect of sampling time is introduced into the algorithms.

##### B. DERIVATION OF THE GENERALIZED TRANSFER RATIO FOR A TWO PORT WAVE DIGITAL FILTER ALGORITHM

Consider the two port network  $N_1$  of Figure 4.1b whose inputs are  $a_1$  and  $a_2$  and outputs are  $b_1$  and  $b_2$ . In order to use the chain matrix theory developed earlier (Chapter II, Section B-D) we need to find port two input and output waves  $a_2, b_2$  in terms of port one input and output waves  $a_1, b_1$  or vice versa.



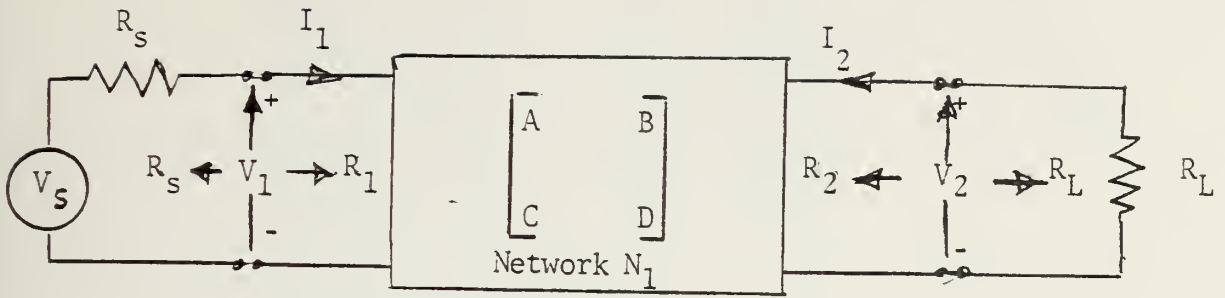


Fig. 4.1a.

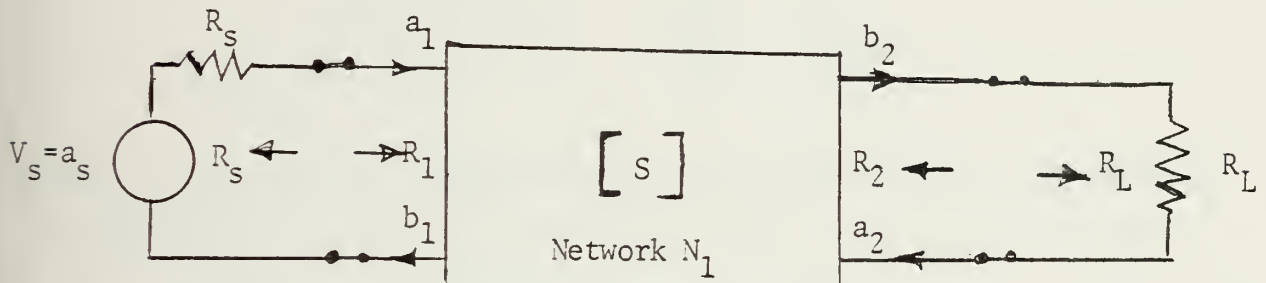


Fig. 4.1b.

Fig. 4.1. Two generalized representations of a two port network. Fig. 4.1a representation in terms of voltage and current quantities. Fig. 4.1b representation in terms of scattering matrix wave quantities. Note that  $R_1$  is the input impedance of the network and  $R_2$  is the output impedance of the network.  $R_S$  is the source impedance and  $R_L$  is the load impedance.



Thus considering the scattering matrix model of Figure 4.1b, we have

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} 1 & R_1 \\ 1 & -R_1 \end{bmatrix} \begin{bmatrix} V_1 \\ I_1 \end{bmatrix} \quad (4.1)$$

and

$$\begin{bmatrix} a_2 \\ b_2 \end{bmatrix} = \begin{bmatrix} 1 & R_2 \\ 1 & -R_2 \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix} \quad (4.2)$$

where  $R_1$  and  $R_2$  are port impedances of port one and port two respectively.

Note that  $a_1, b_1$  and  $a_2, b_2$  are voltage waves. Now in order to find  $a_2, b_2$  in terms of  $a_1, b_1$  we have the chain matrix of

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix} \quad (4.3)$$

Equations (4.1) and (4.3) lead to

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} 1 & R_1 \\ 1 & -R_1 \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix}$$





or

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} A + R_1 C & B + R_1 D \\ A - R_1 C & B - R_1 D \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix} \quad (4.4)$$

But from equation (4.2)

$$\begin{bmatrix} V_2 \\ I_2 \end{bmatrix} = \begin{bmatrix} 1 & R_2 \\ 1 & -R_2 \end{bmatrix}^{-1} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix}$$

or

$$\begin{bmatrix} V_2 \\ I_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2}R_2 & -\frac{1}{2}R_2 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} \quad (4.5)$$

Thus equations (4.4) and (4.5) lead to

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} \frac{A+R_1 C}{2} + \frac{1}{2R_2} (B+R_1 D) & \frac{A+R_1 C}{2} - \frac{1}{2R_2} (B+R_1 D) \\ \frac{A-R_1 C}{2} + \frac{1}{2R_2} (B-R_1 D) & \frac{A-R_1 C}{2} - \frac{1}{2R_2} (B-R_1 D) \end{bmatrix} \begin{bmatrix} b_2 \\ a_2 \end{bmatrix} \quad (4.6)$$

or

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} \mu & \lambda \\ \nu & \kappa \end{bmatrix} \begin{bmatrix} b_2 \\ a_2 \end{bmatrix} \quad (4.7)$$

where

$$\mu = \frac{A+R_1 C}{2} + \frac{1}{2R_2} (B+R_1 D) \quad (4.7a)$$

$$\lambda = \frac{A+R_1 C}{2} - \frac{1}{2R_2} (B+R_1 D) \quad (4.7b)$$



$$v = \frac{A-R_1C}{2} + \frac{1}{2R_2} (B-R_1D) \quad (4.7c)$$

$$\kappa = \frac{A-R_1C}{2} - \frac{1}{2R_2} (B-R_1D) \quad (4.7d)$$

Thus from equation (4.7) we have

$$b_1 = \frac{v}{\mu} a_1 + \left(\kappa - \frac{\lambda v}{\mu}\right) a_2 \quad (4.8)$$

$$b_2 = \frac{1}{\mu} a_1 - \frac{\lambda}{\mu} a_2 \quad (4.9)$$

and the appropriate wave flow diagram is drawn in Figure 4.2.

Note that

$$V_1 = V_s - I_1 R_s \quad (4.10)$$

$$V_2 = - I_2 R_L \quad (4.11)$$

Thus equations (4.2) and (4.11) lead to

$$a_2 = V_2 + R_2 I_2$$

$$b_2 = V_2 - R_2 I_2$$

$$V_2 = - I_2 R_L$$

so that

$$a_2 = b_2 \phi \quad (4.12)$$

$$b_2 = \frac{2V_2}{1+\phi} \quad (4.13)$$

where

$$\phi = \frac{R_L - R_2}{R_L + R_2} \quad (4.13a)$$



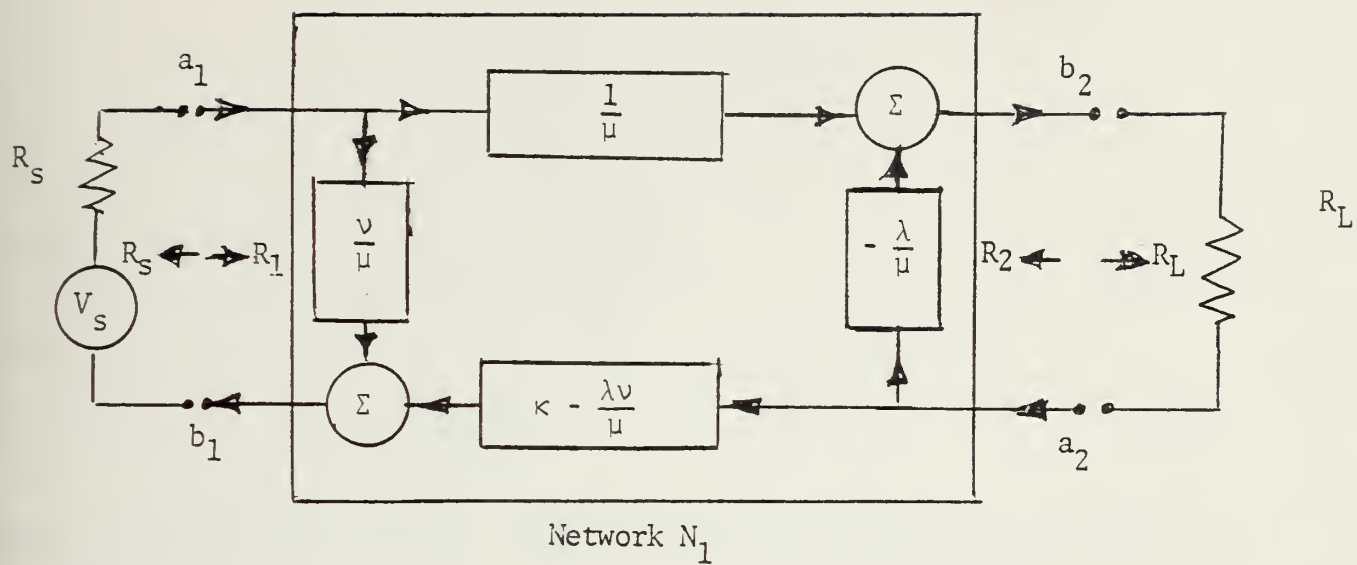


Fig. 4.2. Wave flow diagram representing network  $N_1$  of Fig. 4.1.



and also equations (4.1) and (4.10) lead to

$$a_1 = V_1 + R_1 I_1$$

$$b_1 = V_1 - R_1 I_1$$

$$V_1 = V_s - I_1 R_s$$

so that

$$a_1 + \theta b_1 = (1 + \theta) V_s \quad (4.14)$$

where

$$\theta = \frac{R_1 - R_s}{R_1 + R_s} \quad (4.15)$$

Where  $V_s$  is the input voltage. Thus from equations (4.13) and (4.14) we have

$$H(z) = \frac{V_2}{V_s} = \frac{(1+\phi)(1+\theta)}{2} \cdot \frac{b_2(z)}{a_1(z) + \theta b_1(z)} \quad (4.16)$$

Note that for  $\theta = 0$ , from equation (4.15),  $R_1 = R_s$ .

$$V_s = a_s = a_1$$

and

$$H(z) = \frac{(1+\phi)}{2} \cdot \frac{b_2(z)}{a_s(z)} \quad (4.17)$$

For  $\phi = 0$ , from equation (4.13a),  $R_2 = R_L$  and from equation (4.12)  $a_2 = 0$ .

Thus from equation (4.16)

$$H(z) = \frac{(1+\theta)}{2} \cdot \frac{b_2(z)}{a_s(z) + \theta b_1(z)} \quad (4.18)$$

or

$$H(z) = \frac{1+\theta}{2(1 + \theta \frac{V}{\mu})} \cdot \frac{b_2(z)}{a_s(z)} \quad (4.19)$$





The general realization with the aid of equations (4.11), (4.14) and (4.17) is shown in Figure 4.3.

### C. REALIZATION OF TWO PORT WAVE DIGITAL STRUCTURE

Theoretically LC passive structures such as ladder, lattice and most of the other symmetrical structures can be reduced to a form of Figure 4.4 which is the "T" form with positive or negative element values. Note that in A B C D matrix, the direction if  $I_2$  is in the direction as shown in Figure 4.4 and  $Z_1$ ,  $Z_2$  or  $Z_3$  can be positive or negative capacitors, inductors and/or tank circuits with positive or negative element values.

The A B C D matrix of Figure 4.4 as per equation (4.3) is found from the equations

$$V_1 = V_2 + I_2 Z_3 + I_1 Z_1$$

$$(I_1 - I_2) Z_2 = V_2 + I_2 Z_3$$

and the A B C D matrix is

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} 1 + \frac{Z_1}{Z_2} & Z_1 + Z_3 + \frac{Z_1 Z_3}{Z_2} \\ \frac{1}{Z_2} & 1 + \frac{Z_3}{Z_2} \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix} \quad (4.20)$$

Thus

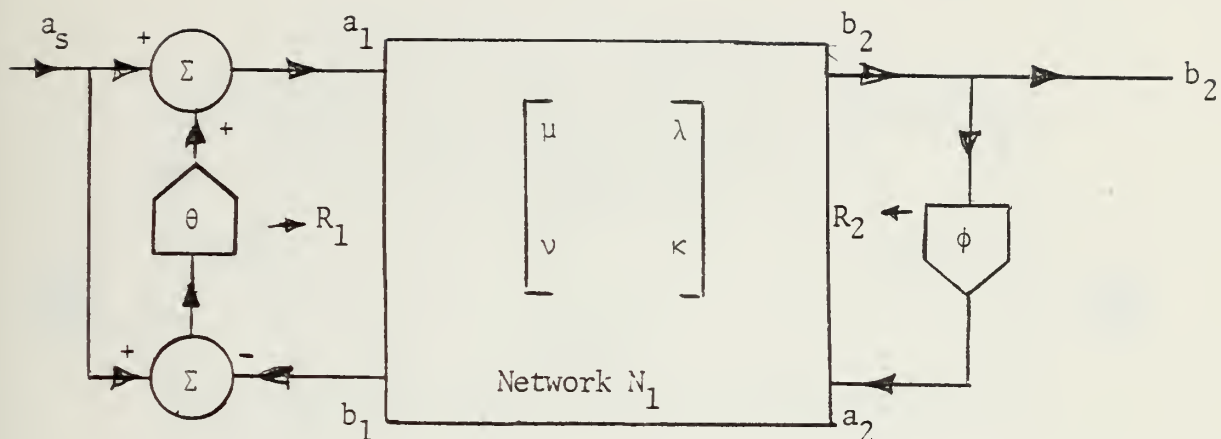
$$A = 1 + \frac{Z_1}{Z_2} \quad (4.20a)$$

$$B = Z_1 + Z_3 + \frac{Z_1 Z_3}{Z_2} \quad (4.20b)$$

$$C = \frac{1}{Z_2} \quad (4.20c)$$

$$D = 1 + \frac{Z_3}{Z_2} \quad (4.20d)$$





$$\theta = \frac{R_1 - R_s}{R_1 + R_s}$$

$$\phi = \frac{R_L - R_2}{R_L + R_2}$$

Fig. 4.3. General realization of a two port network  $N_1$  with input and output waves. Note that  $a_s$  is the input voltage wave.



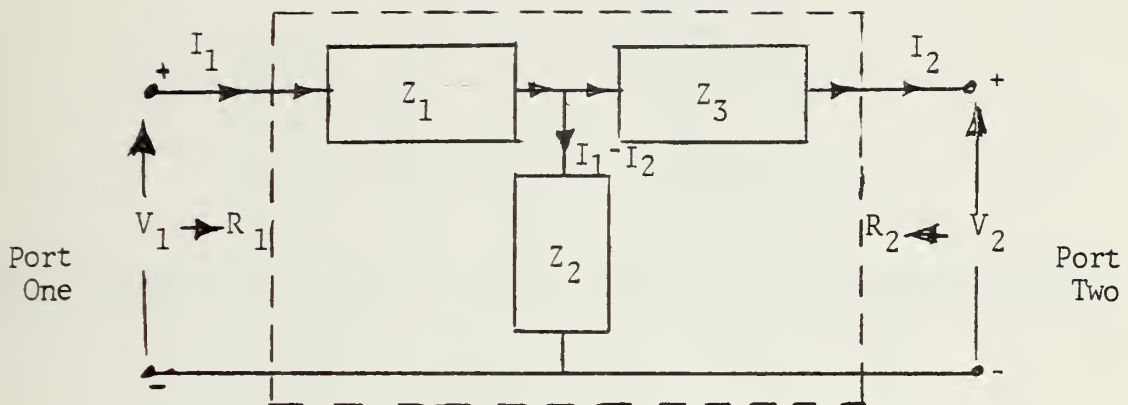


Fig. 4.4. A general two port "T" network with complex elements  $Z_1$ ,  $Z_2$ ,  $Z_3$ . Note that  $R_1$  is the input impedance of Port 1, and  $R_2$  is the input impedance of Port 2.



Note that for a single series element  $Z_1$  as per Figure 4.5 and equation (4.20)

$$Z_3 = 0$$

$$Z_2 = \infty$$

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} 1 & Z_1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix}$$

and for a single shunt element  $Z_2$  as per Figure 4.6 and equation (4.20)

$$Z_1 = 0$$

$$Z_3 = 0$$

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{1}{Z_2} & 1 \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix}$$

Now we can easily find the values of  $\nu$ ,  $\lambda$ ,  $\mu$ ,  $\kappa$  of equation (4.7).

Thus equations (4.7a) and (4.20) lead to

$$\mu = \frac{R_1 + R_2 + Z_1 + Z_3 + \frac{(R_1 + Z_1)(R_2 + Z_3)}{Z_2}}{2R_2} \quad (4.21)$$

and equations (4.7b) and (4.20) lead to

$$\lambda = \frac{-R_1 + R_2 - Z_1 - Z_3 + \frac{(R_1 + Z_1)(R_2 - Z_3)}{Z_2}}{2R_2} \quad (4.22)$$





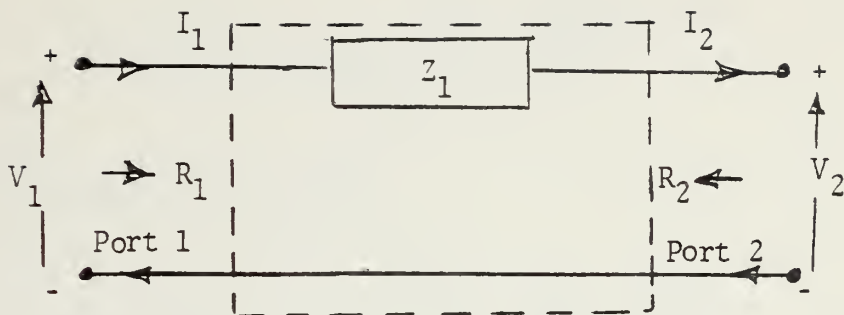


Fig. 4.5. A general two port with series element only.  
Note that  $R_1$  and  $R_2$  are input impedances of Port 1 and Port 2 respectively.

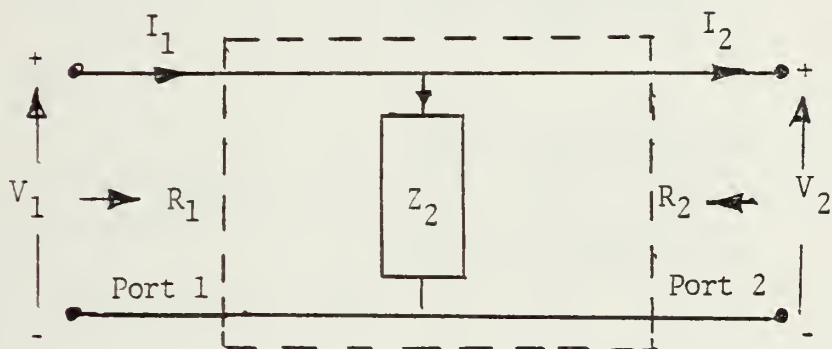


Fig. 4.6. A general two port with shunt element only.  
Note that  $R_1$  and  $R_2$  are input impedances of Port 1 and Port 2 respectively.



equations (4.7c) and (4.20) lead to

$$v = \frac{-R_1 + R_2 + Z_1 + Z_3 + \frac{(-R_1 + Z_1)(R_2 + Z_3)}{Z_2}}{2R_2} \quad (4.23)$$

equations (4.7d) and (4.20) lead to

$$k = \frac{R_1 + R_2 - Z_1 + Z_3 + \frac{(Z_1 - R_1)(R_2 - Z_3)}{Z_2}}{2R_2} \quad (4.24)$$

Thus the elements of Figure 4.2 will be

$$\frac{1}{\mu} = \frac{2R_2}{R_1 + R_2 + Z_1 + Z_3 + \frac{(R_1 + Z_1)(R_2 + Z_3)}{Z_2}} \quad (4.25)$$

$$\frac{v}{\mu} = \frac{-R_1 + R_2 + Z_1 + Z_3 + \frac{(-R_1 + Z_1)(R_2 + Z_3)}{Z_2}}{R_1 + R_2 + Z_1 + Z_3 + \frac{(R_1 + Z_1)(R_2 + Z_3)}{Z_2}} \quad (4.26)$$

$$k - \frac{\lambda v}{\mu} = \frac{2R_1}{R_1 + R_2 + Z_1 + Z_3 + \frac{(R_1 + Z_1)(R_2 - Z_3)}{Z_2}} \quad (4.27)$$

$$- \frac{\lambda}{\mu} = \frac{R_1 - R_2 + Z_1 + Z_3 - \frac{(R_1 + Z_1)(R_2 - Z_3)}{Z_2}}{R_1 + R_2 + Z_1 + Z_3 + \frac{(R_1 + Z_1)(R_2 - Z_3)}{Z_2}} \quad (4.28)$$

Note that for the single series element of Figure 4.5 the equations (4.25),

(4.26), (4.27), (4.28) reduce to



$$\frac{1}{\mu} = \frac{2R_2}{R_1 + R_2 + Z_1} \quad (4.29)$$

$$\frac{v}{\mu} = \frac{-R_1 + R_2 + Z_1}{R_1 + R_2 + Z_1} \quad (4.30)$$

$$\kappa - \frac{\lambda v}{\mu} = \frac{2R_1}{R_1 + R_2 + Z_1} \quad (4.31)$$

$$- \frac{\lambda}{\mu} = \frac{R_1 - R_2 + Z_1}{R_1 + R_2 + Z_1} \quad (4.32)$$

and for a single shunt element of Figure 4.6 the aforesaid equations will be

$$\frac{1}{\mu} = \frac{2G_1}{G_1 + G_2 + Y_2} \quad (4.33)$$

$$\frac{v}{\mu} = \frac{-Y_2 - G_1 + G_2}{Y_2 + G_1 + G_2} \quad (4.34)$$

$$\kappa - \frac{\lambda v}{\mu} = \frac{2G_2}{Y_2 + G_1 + G_2} \quad (4.35)$$

$$- \frac{\lambda}{\mu} = \frac{-Y_2 + G_1 - G_2}{Y_2 + G_1 + G_2} \quad (4.36)$$

where

$$G_1 = \frac{1}{R_1}$$

and

$$G_2 = \frac{1}{R_2}$$

$$Y_2 = \frac{1}{Z_2}$$



D. DERIVATION OF GENERALIZED ALGORITHM WITH NO DELAY FREE PATH IN PORT ONE OR SIMILARLY WITH NO DELAY FREE PATH IN PORT TWO

The wave flow diagram of Figure 4.2 is not suitable to be used in the chain matrix equation, since most probably it has delay free paths between  $a_1$  to  $b_1$  and  $a_2$  to  $b_2$  which we have not analyzed as yet, and if so it is not desirable. Thus by a careful choice of either  $R_1$  or  $R_2$  we can force the wave flow from either  $a_1$  to  $b_1$  to have no delay free path, or from  $a_2$  to  $b_2$  to have no delay free path; but not both of them since we have only one variable to adjust, and that is either  $R_1$  or  $R_2$ .

The delay free wave flow for the case of no delay free path from  $a_1$  to  $b_1$  for a three section filter is shown in Figure 4.7 and the delay free wave flow for the case of no delay free path from  $a_2$  to  $b_2$  for the same section filter is shown in Figure 4.8.

Note that a close examination of Figure 4.7 and Figure 4.8 still reveals a delay free feedback loop at the terminating points. Thus to make the sections of practical value,  $\phi$  must be forced equal to zero for Figure 4.7 implying that

$$\phi = \frac{R_L - R_2}{R_L + R_2} = 0 \quad \text{or} \quad R_L = R_2 \quad (4.37)$$

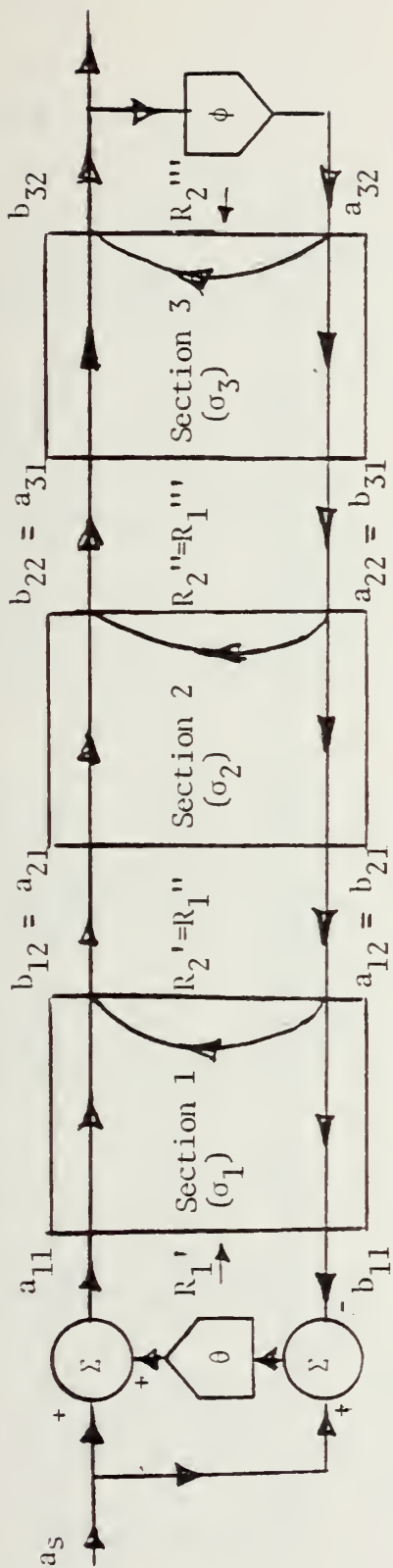
and for the Figure 4.8,  $\theta$  must be forced equal to zero implying that

$$\theta = \frac{R_1''' - R_S}{R_1''' + R_S} = 0 \quad \text{or} \quad R_S = R_1''' \quad (4.38)$$

Thus the updated and useful delay free wave flow diagrams would be as per Figures 4.9 and 4.10. Note that in the first case we have to start at load end and work our way through to the source end, and in the second





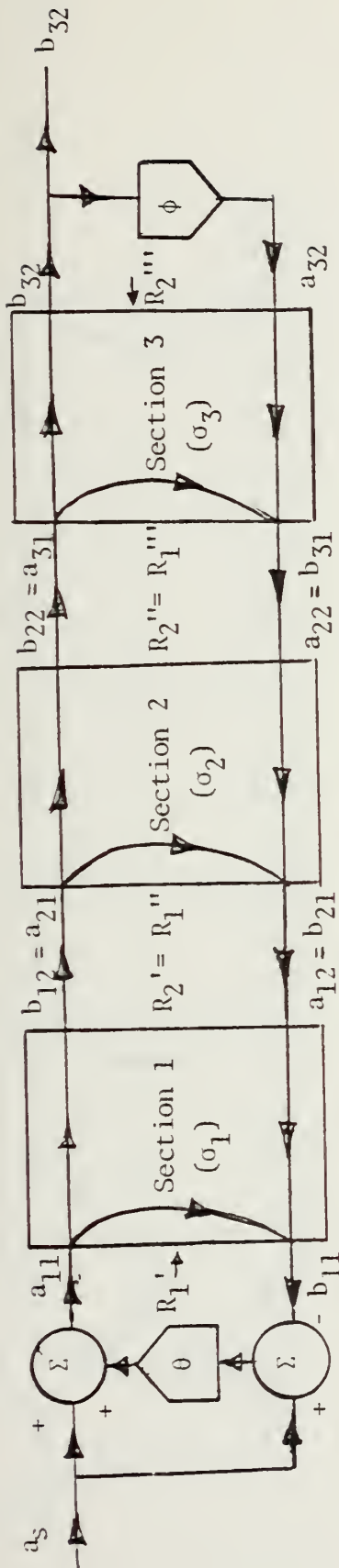


$$\theta = \frac{R_S - R_1'}{R_S + R_1'}$$

$$\phi = \frac{R_L - R_2'''}{R_L + R_2'''}$$

Figure 4.7. Signal flow graph of the direct delay free paths for three cascaded two ports with no delay free path from  $a_1$  to  $b_1$ , with multiplier coefficients  $\sigma_1$ ,  $\sigma_2$ , and  $\sigma_3$ . Note that this is not a complete signal flow graph since delayed signal paths are not shown. Note also that  $R_1'$  is the input impedance of the port one of the section 1, and  $R_2'''$  is the input impedance of the port two of the section 3.



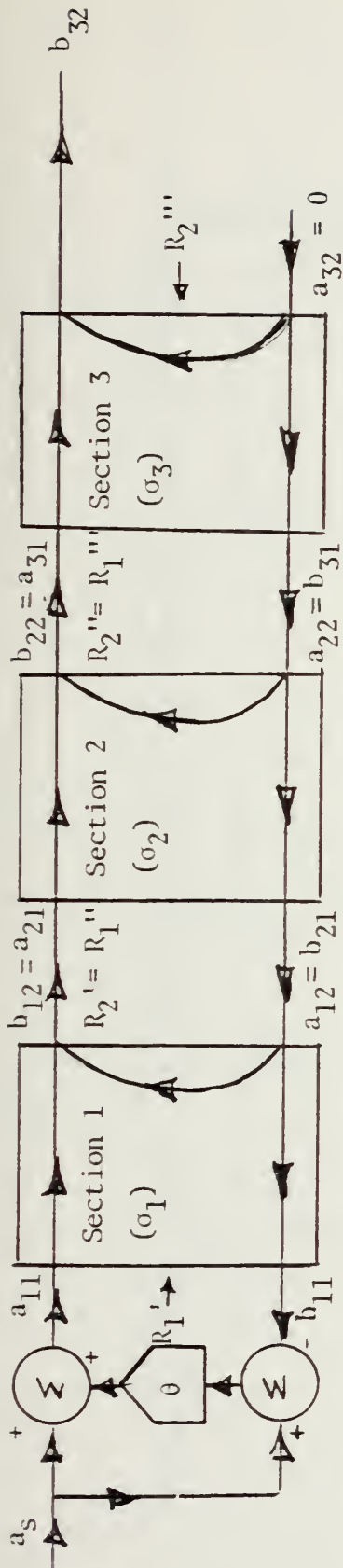


$$\theta = \frac{R_s - R_1'}{R_s + R_1'}$$

$$\phi = \frac{R_L - R_2'''}{R_L + R_2'''}$$

Fig. 4.8. Signal flow graph of the direct delay free paths for a three cascaded two ports with no delay free path from  $a_2$  to  $b_2$ . Note that this is not a complete signal flow graph since the delayed signal paths are not shown. Note also that  $R_1'$  is the input impedance of the port one of the section 1, and  $R_2'''$  is the input impedance of the port two of the section 3.





$$\theta = \frac{R_S - R_L'}{R_S + R_L'}$$

$$R_L = R_2'''$$

Figure 4.9. Signal flow graph of Fig. 4.7 modified to eliminate the existing delay free loop at the terminating load (no delay free path from  $a_1$  to  $b_1$ ).









case, i.e. Fig. 4.10, we have to start at the source and work through to the load end. Alternately we can start from both ends and progress towards each other. Where they meet we have to introduce a transformer or unit element such as Kuroda's identity, but this is not really recommended since it introduces further complications. This approach is not desirable and is to be avoided.

Now with the aid of formulas (4.25), (4.26), (4.27), (4.28), we can cascade as many elements in a section as complex as Figure 4.4 or as simple as Figure 4.5 or Figure 4.6.

Note that the element values  $Z_1$ ,  $Z_2$  and  $Z_3$  must be in s domain, and for digitization we use the bilinear transformation

$$s = \frac{2}{T} \frac{(z-1)}{(z+1)} \quad (4.39)$$

where T is the sampling period of the digital filter.

#### Example 4-1

Series L, no delay free path from  $a_1$  to  $b_1$ , the case of a simple element

With reference to Figure 4.2,  $v/\mu$  must have no delay free path, and the aim is to find  $R_1$  given  $R_2$ , the load end terminating resistance. From equation (4.30)

$$\frac{v}{\mu} = \frac{Z_1 - R_1 + R_2}{Z_1 + R_1 + R_2}$$

and

$$Z_1 = sL = \frac{2}{T} \frac{(z-1)}{(z+1)} \cdot L$$

Thus

$$\frac{v}{\mu} = \frac{z \cdot \left( \frac{2L}{T} + R_2 - R_1 \right) + (R_2 - R_1 - \frac{2L}{T})}{z \cdot \left( \frac{2L}{T} + R_1 + R_2 \right) + (R_1 + R_2 - \frac{2L}{T})} \quad (4.40)$$



for  $\frac{v}{u}$  to be with no delay free path from equation (4.40)

$$\frac{2L}{T} + R_2 - R_1 = 0$$

which leads to 
$$R_1 = R_2 + \frac{2L}{T} \quad (4.41)$$

Thus from equations (4.29), (4.30), (4.31), (4.32), (4.38), (4.39), (4.41) we have

$$\frac{v}{u} = \frac{\sigma-1}{z+\sigma} \quad \text{where} \quad \sigma = \frac{R_2}{R_1} \quad (4.42)$$

$$\begin{aligned} \frac{1}{u} &= \frac{2R_2}{z_1 + R_1 + R_2} \\ &= \frac{\sigma(z+1)}{(z+\sigma)} \end{aligned} \quad (4.43)$$

$$\begin{aligned} \kappa - \frac{\lambda v}{u} &= \frac{2R_1}{R_1 + R_2 + z_1} \\ &= \frac{z+1}{z+\sigma} \end{aligned} \quad (4.44)$$

$$\begin{aligned} -\frac{\lambda}{v} &= \frac{R_1 - R_2 + z_1}{R_1 + R_2 + z_1} \\ &= \frac{z(1-\sigma)}{z+\sigma} \end{aligned} \quad (4.45)$$

Thus the output equations for series L with no delay free path from  $a_1$  to  $b_1$  are

$$\begin{aligned} b_1 &= \frac{\sigma-1}{z+\sigma} a_1 + \frac{z+1}{z+\sigma} a_2 \\ b_2 &= \frac{\sigma(z+1)}{z+\sigma} a_1 + \frac{z(1-\sigma)}{z+\sigma} a_2 \end{aligned} \quad (4.46)$$

or

$$\begin{aligned} b_1 &= \frac{(\sigma-1)z^{-1}}{1 + \sigma z^{-1}} a_1 + \frac{1+z^{-1}}{1+\sigma z^{-1}} a_2 \\ b_2 &= \frac{\sigma(1+z^{-1})}{1 + \sigma z^{-1}} a_1 + \frac{(1-\sigma)}{1+\sigma z^{-1}} a_2 \end{aligned} \quad (4.47)$$



with

$$\sigma = \frac{R_2}{R_1}$$

and the iterative equations, corresponding to (4.46), (4.47) are

$$b_1(n) = (\sigma-1)a_1(n-1) + a_2(n) + a_2(n-1) - \sigma b_1(n-1) \quad (4.48)$$

$$b_2(n) = \sigma a_1(n) + \sigma a_1(n-1) + (1-\sigma)a_2(n) - \sigma b_2(n-1) \quad (4.49)$$

With zero initial conditions, i.e.

$$x_1 = 0$$

$$x_2 = 0$$

$$x_3 = 0$$

$$x_4 = 0$$

The equations (4.48), (4.49) can be rewritten as

$$b_1 = \sigma(x_1 - x_3) - x_1 - x_2 + a_2 \quad (4.50)$$

$$b_2 = \sigma(a_1 - a_2 + x_1 - x_4) + a_2 \quad (4.51)$$

and updated equations

$$x_1 = a_1 \quad (4.52)$$

$$x_2 = b_1 \quad (4.53)$$

$$x_3 = a_2 \quad (4.54)$$

$$x_4 = b_2 \quad (4.55)$$

Note that there is only two multiplications for each iteration.



### Example 4-2

An example of designing the composite series and shunt element with no delay free path on port two.

As an example of a composite series and shunt algorithm, let's consider the two element section of Fig. 4.11. The aim is to derive the wave flow algorithm with no delay free path from  $a_2$  to  $b_2$ . From equation (4.28) we have

$$\begin{aligned}
 -\frac{\lambda}{\mu} &= \frac{R_1 - R_2 + sL - (R_1 + sL) R_2 sC}{R_1 + R_2 + sL + (R_1 + sL) R_2 sC} \\
 &= \frac{R_1 - R_2 + s(L - R_1 R_2 C) - s^2 (R_2 LC)}{R_1 + R_2 + s(L + R_1 R_2 C) + s^2 (R_2 LC)} \quad \left| \quad s = \frac{2}{T} \frac{(z-1)}{(z+1)} \right. \\
 &= \frac{A_2 z^2 + B_2 z + C_2}{A_1 z^2 + B_1 z + C_1}
 \end{aligned}$$

where

$$A_2 = R_1 - R_2 + \frac{2L}{T} - \frac{2R_1 R_2 C}{T} - \frac{4R_2 LC}{T^2}$$

$$B_2 = 2(R_1 - R_2 + \frac{4R_2 LC}{T^2})$$

$$C_2 = R_1 - R_2 - \frac{2L}{T} + \frac{2R_1 R_2 C}{T} - \frac{4R_2 LC}{T^2}$$

$$A_1 = R_1 + R_2 + \frac{2L}{T} + \frac{2R_1 R_2 C}{T} + \frac{4R_2 LC}{T^2}$$

$$B_1 = 2(R_1 + R_2 - \frac{4R_2 LC}{T^2})$$

$$C_1 = R_1 + R_2 + \frac{2L}{T} - \frac{2R_1 R_2 C}{T} + \frac{4R_2 LC}{T^2}$$

with reference to Fig. 4.2 for no delay free path on port two,  $-\frac{\lambda}{\mu}$  must have no delay free path, thus

$$A_2 = 0$$





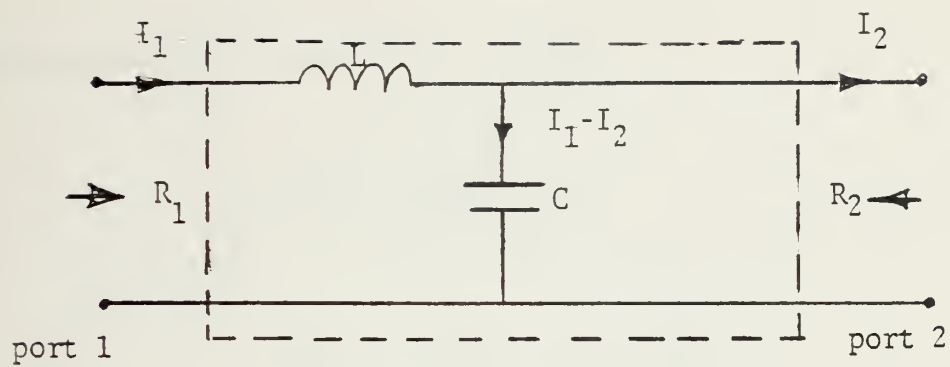


Figure 4.11. Composite series and shunt element of example 4.2.



This leads to

$$R_2 = \frac{R_1 + \frac{2L}{T}}{1 + \frac{2R_1 C}{T} + \frac{4LC}{T^2}} \quad (4.57)$$

Thus from equations (4.57) and (4.28) we have

$$-\frac{\lambda}{\mu} = \frac{\beta_2 z^{-1} + \gamma_2 z^{-2}}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} \quad (4.58)$$

where

$$\beta_2 = \frac{\frac{2R_1^2 C}{T} + \frac{8R_1 LC}{T^2} - \frac{2L}{T} + \frac{8L^2 C}{T^3}}{(R_1 + \frac{2L}{T})(1 + \frac{2R_1 C}{T} + \frac{4LC}{T^2})}$$

$$\gamma_2 = \frac{\frac{2R_1^2 C}{T} - \frac{2L}{T} - \frac{8L^2 C}{T^3}}{(R_1 + \frac{2L}{T})(1 + \frac{2R_1 C}{T} + \frac{4LC}{T^2})}$$

$$\beta_1 = \frac{2R_1 + \frac{2L}{T} + \frac{2R_1^2 C}{T} - \frac{8L^2 C}{T^3}}{(R_1 + \frac{2L}{T})(1 + \frac{2R_1 C}{T} + \frac{4LC}{T^2})}$$

$$\gamma_1 = \frac{R_1}{(R_1 + \frac{2L}{T})(1 + \frac{2R_1 C}{T} + \frac{4LC}{T^2})}$$

and from equations (4.57) and (4.25) we have

$$\frac{1}{\mu} = \frac{(1+z^{-1})^2 \beta_3}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} \quad (4.59)$$

where

$$\beta_3 = \frac{1}{1 + \frac{2R_1 C}{T} + \frac{4LC}{T^2}}$$



and from equations (4.57) and (4.27) we have

$$\kappa - \frac{\lambda v}{\mu} = \frac{\beta_4 (1+z^{-1})^2}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} \quad (4.60)$$

where

$$\beta_4 = \frac{R_1}{R_1 + \frac{2L}{T}}$$

and from equations (4.57) and (4.26) we have

$$\frac{v}{\mu} = \frac{-\beta_2 z^{-1} - \gamma_2}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} \quad (4.61)$$

Thus the wave flow algorithms for the composite element with no delay free path from  $a_2$  to  $b_2$  will be

$$b_2 = \frac{\beta_3 (1+z^{-1})^2}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} a_1 + \frac{\beta_2 z^{-1} + \gamma_2 z^{-2}}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} a_2 \quad (4.62)$$

$$b_1 = \frac{-\beta_2 z^{-1} - \gamma_2}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} a_1 + \frac{\beta_4 (1+z^{-1})^2}{1 + \beta_1 z^{-1} + \gamma_1 z^{-2}} a_2 \quad (4.63)$$

or

$$\begin{aligned} b_2(n) &= \beta_3 a_1(n) + 2\beta_3 a_1(n-1) + \beta_3 a_1(n-2) + \beta_2 a_2(n-1) \\ &\quad + \gamma_2 a_2(n-2) - \beta_1 b_2(n-1) - \gamma_1 b_2(n-2) \end{aligned} \quad (4.64)$$



$$b_1(n) = -\beta_2 a_1(n-1) - \gamma_2 a_1(n) + \beta_4 a_2(n) + 2\beta_4 a_2(n-1) + \beta_4 a_2(n-2) \\ - \beta_1 b_1(n-1) - \gamma_1 b_1(n-2) \quad (4.65)$$

and the iterative equation corresponding to equations (4.62), (4.63) with initial conditions set to zero, i.e.

$$x_i = 0 \quad i=1,8$$

will lead to the iterative equations of

$$b_2 = \beta_3(a_1 + 2x_1 + x_2) + \beta_2 x_3 + \gamma_2 x_4 - \beta_1 x_7 - \gamma_1 x_8 \quad (4.66)$$

$$b_1 = -\gamma_2 a_1 - \beta_2 x_1 + \beta_4(a_2 + 2x_3 + x_4) - \beta_1 x_5 - \gamma_1 x_6 \quad (4.67)$$

with updated equations of

$$\begin{aligned} x_1 &= a_1 \\ x_2 &= x_1 \\ x_3 &= a_2 \\ x_4 &= x_3 \\ x_5 &= b_1 \\ x_6 &= x_5 \\ x_7 &= b_2 \\ x_8 &= x_7 \end{aligned} \quad (4.68)$$





Note that there are 12 multiplications with six coefficients, i.e.  $\alpha$ 's,  $\beta$ 's,  $\gamma$ 's. It is interesting to note that since in this example we originally had two elements, i.e. Series L, and shunt C with input impedance  $R_1$ . Thus we expect the  $\alpha$ 's,  $\beta$ 's, and  $\gamma$ 's to be functions of  $R_1$ , L, and C only. This can be easily shown by identifying two new variables  $\alpha$  and  $\beta$  such that

$$\alpha = \frac{\frac{2L}{T}}{R_1 + \frac{2L}{T}}$$

$$\beta = \frac{1}{1 + \frac{2R_1 C}{T} + \frac{4LC}{T^2}}$$

Then

$$R_2^* = \frac{L\beta}{\alpha} = R_1 \cdot \frac{(\beta)}{1-\alpha}$$

$$\beta_1 = 1 - 2\alpha + \beta(\alpha+1)$$

$$\gamma_1 = \beta(1-\alpha)$$

$$\beta_2 = 1 - \beta(1+\alpha)$$

$$\gamma_2 = 1 - 2\alpha + \beta(\alpha-1)$$

$$\beta_3 = \beta$$

$$\beta_4 = 1-\alpha$$

---

\*  $R_2 = \frac{L\beta}{\alpha}$  is more general even for the case  $R_1=R_S=0$  which makes  $\alpha=1$  and  $1-\alpha=0$  and if we use  $R_2 = R_1 \frac{(\beta)}{1-\alpha}$ ,  $R_2$  becomes  $\frac{0}{0}$  which is indefinite, while  $R_2 = \frac{L\beta}{\alpha}$  is not indefinite.



By inserting these new values into the iteration equations (4.66) and (4.67) the number of multiplications reduces from 12 to 9 which is significant. Thus the new iterative equations from (4.66), (4.67) will be

$$b_1 = a_2 - a_1 - x_1 + 2x_3 + x_4 - x_5 + \alpha(-a_2 - x_4 + 2(a_1 - x_3 + x_5)) \quad (4.69) \\ + \beta(a_1 + x_1 - x_5 - x_6) + \alpha\beta(-a_1 + x_1 - x_5 + x_6)$$

$$b_2 = x_3 + x_4 - x_7 + 2\alpha(x_7 - x_4) + \beta(a_1 + 2x_1 + x_2 - x_3 - x_4 - x_7 - x_8) \quad (4.70) \\ + \alpha\beta(-x_3 + x_4 - x_7 + x_8)$$

Note that the multiplication  $\alpha\beta$  and  $2\alpha$  does not enter into the iteration and they can be premultiplied before the starting of the iteration process.

#### Notation

For latter analysis in Chapter V we call these latter set of equations i.e. equations (4.69) and (4.70) as "reduced parameter complex wave digital algorithms" and the original set, i.e. equations (4.66) and (4.67) as "complex wave digital algorithms".

Although theoretically it is always possible to find new variables equal to the number of original L's and C's, in practice it becomes a tedious job for more than two variables, and it is not recommended.

#### E. TABULATION OF SIMPLE L AND C ELEMENTS IN SERIES AND SHUNT

In a similar way as examples 1 and 2, the wave flow equations for the single L and C elements for both cases of series and shunt are tabulated in Tables 4.1a,b and 4.2a,b.

Note that Tables 4.1a, 4.1b are for the case of no delay free path from  $a_1$  to  $b_1$ , and Tables 4.2a, 4.2b are for the case of no delay free path from  $a_2$  to  $b_2$ .



Table 4.1a. Algorithms for simple L and C elements with no delay free path from  $a_1$  to  $b_1$  with sampling time T.

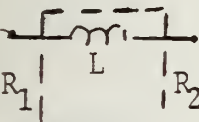



Element	Algorithm	Remarks
 <p>(1) (2) Series L</p>	$b_1 = \frac{\sigma-1}{z+\sigma} a_1 + \frac{z+1}{z+\sigma} a_2$ $b_2 = \frac{\sigma(z+1)}{z+\sigma} a_1 + \frac{z(1-\sigma)}{z+\sigma} a_2$	$R_1 = R_2 + \frac{2L}{T}$ $\sigma = \frac{R_2}{R_1}$
 <p>(1) (2) Series C</p>	$b_1 = \frac{1-\sigma}{z-\sigma} a_1 + \frac{z-1}{z-\sigma} a_2$ $b_2 = \frac{\sigma(z-1)}{z-\sigma} a_1 + \frac{z(1-\sigma)}{z-\sigma} a_2$	$R_1 = R_2 + \frac{T}{2C}$ $\sigma = \frac{R_2}{R_1}$
 <p>(1) (2) Shunt L</p>	$b_1 = \frac{\sigma-1}{z-\sigma} a_1 + \frac{\sigma(z-1)}{z-\sigma} a_2$ $b_2 = \frac{(z-1)}{z-\sigma} a_1 + \frac{z(\sigma-1)}{z-\sigma} a_2$	$G_1 = G_2 + \frac{T}{2L}$ $\sigma = \frac{R_1}{R_2} = \frac{G_2}{G_1}$
 <p>(1) (2) Shunt C</p>	$b_1 = \frac{1-\sigma}{z+\sigma} a_1 + \frac{\sigma(z+1)}{z+\sigma} a_2$ $b_2 = \frac{z+1}{z+\sigma} a_1 + \frac{z(\sigma-1)}{z+\sigma} a_2$	$G_1 = G_2 + \frac{2C}{T}$ $\sigma = \frac{R_1}{R_2} = \frac{G_2}{G_1}$



Table 4.1b. Iterative algorithms for simple L and C elements with no delay free path from  $a_1$  to  $b_1$  with sampling time T and with initial conditions i.e.  $x_1, x_2, x_3, x_4$  set to zero

Element	Iterative Algorithm	Remarks	Updated Values
<p>Series L</p>	$b_1 = a_2 - x_1 + x_2 + \sigma(x_1 - x_3)$ $b_2 = a_2 + \sigma(a_1 - a_2 + x_1 - x_4)$	$R_1 = R_2 + \frac{2L}{T}$ $\sigma = \frac{R_2}{R_1}$	$x_1 = a_1$ $x_2 = a_2$ $x_3 = b_1$ $x_4 = b_2$
<p>Series C</p>	$b_1 = a_2 + x_1 - x_2 - \sigma(x_1 - x_3)$ $b_2 = a_2 + \sigma(a_1 - a_2 - x_1 + x_4)$	$R_1 = R_2 + \frac{T}{2C}$ $\sigma = \frac{R_2}{R_1}$	
<p>Shunt L</p>	$b_1 = -x_1 + \sigma(a_2 + x_1 - x_2 + x_3)$ $b_2 = a_1 - a_2 - x_1 + \sigma(a_2 + x_4)$	$G_1 = G_2 + \frac{T}{2L}$ $\sigma = \frac{R_1}{R_2} = \frac{G_2}{G_1}$	
<p>Shunt C</p>	$b_1 = x_1 + \sigma(a_2 - x_1 + x_2 - x_3)$ $b_2 = a_1 - a_2 + x_1 + \sigma(a_2 - x_4)$	$G_1 = G_2 + \frac{2C}{T}$ $\sigma = \frac{R_1}{R_2} = \frac{G_2}{G_1}$	





Table 4.2a. Algorithms for simple L and C elements with no delay free path from  $a_2$  to  $b_2$  with sampling time T

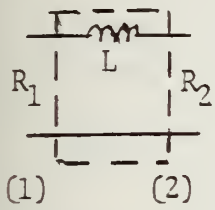
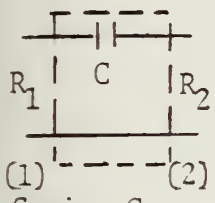
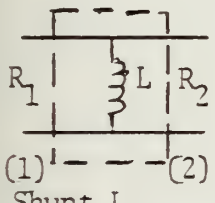
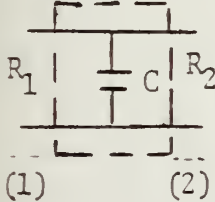
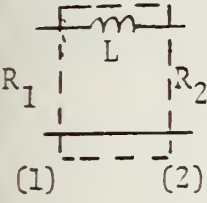
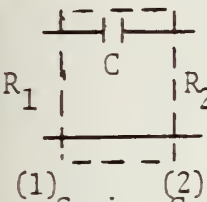
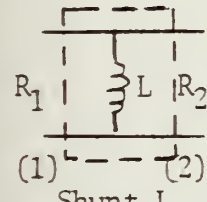
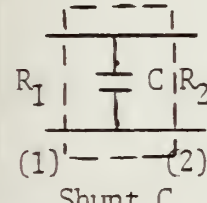
Element	Algorithm	Remarks
 <p>(1) (2) Series L</p>	$b_1 = \frac{z(1-\sigma)}{z+\sigma} a_1 + \frac{\sigma(z+1)}{z+\sigma} a_2$ $b_2 = \frac{z+1}{z+\sigma} a_1 + \frac{\sigma-1}{z+\sigma} a_2$	$R_2 = R_1 + \frac{2L}{T}$ $\sigma = \frac{R_1}{R_2}$
 <p>(1) (2) Series C</p>	$b_1 = \frac{z(1-\sigma)}{z-\sigma} a_1 + \frac{\sigma(z-1)}{z-\sigma} a_2$ $b_2 = \frac{z-1}{z-\sigma} a_1 + \frac{(1-\sigma)}{z-\sigma} a_2$	$R_2 = R_1 + \frac{T}{2C}$ $\sigma = \frac{R_1}{R_2}$
 <p>(1) (2) Shunt L</p>	$b_1 = \frac{z(\sigma-1)}{z-\sigma} a_1 + \frac{z-1}{z-\sigma} a_2$ $b_2 = \frac{\sigma(z-1)}{z-\sigma} a_1 + \frac{\sigma-1}{z-\sigma} a_2$	$G_2 = G_1 + \frac{T}{2L}$ $\sigma = \frac{G_1}{G_2} = \frac{R_1}{R_2}$
 <p>(1) (2) Shunt C</p>	$b_1 = \frac{z(\sigma-1)}{z+\sigma} a_1 + \frac{z+1}{z+\sigma} a_2$ $b_2 = \frac{\sigma(z+1)}{z+\sigma} a_1 + \frac{(1-\sigma)}{z+\sigma} a_2$	$G_2 = G_1 + \frac{2C}{T}$ $\sigma = \frac{G_1}{G_2} = \frac{R_1}{R_2}$



Table 4.2b. Iterative algorithms for simple L and C elements with no delay free path from  $a_2$  to  $b_2$  with sampling time T and with the initial conditions i.e.  $x_1, x_2, x_3, x_4$  set to zero

Element	Iterative Algorithm	Remarks	Updated Values
 <p>Series L</p>	$b_1 = a_1 + \sigma(a_2 - a_1 + x_2 - x_3)$ $b_2 = a_1 + x_1 - x_2 + \sigma(x_2 - x_4)$	$R_2 = R_1 + \frac{2L}{T}$ $\sigma = \frac{R_1}{R_2}$	$x_1 = a_1$ $x_2 = a_2$ $x_3 = b_1$ $x_4 = b_2$
 <p>Series C</p>	$b_1 = a_1 + \sigma(a_2 - a_1 - x_2 + x_3)$ $b_2 = a_1 - x_1 + x_2 - \sigma(x_2 - x_4)$	$R_2 = R_1 + \frac{T}{2C}$ $\sigma = \frac{R_1}{R_2}$	
 <p>Shunt L</p>	$b_1 = a_2 - a_1 - x_2 + \sigma(a_1 + x_3)$ $b_2 = -x_2 + \sigma(a_1 - x_1 + x_2 + x_4)$	$G_2 = G_1 + \frac{T}{2L}$ $\sigma = \frac{G_1}{G_2} = \frac{R_1}{R_2}$	
 <p>Shunt C</p>	$b_1 = a_2 - a_1 + x_2 + \sigma(a_1 - x_3)$ $b_2 = x_2 + \sigma(a_1 + x_1 - x_2 - x_4)$	$G_2 = G_1 + \frac{2C}{T}$ $\sigma = \frac{G_1}{G_2} = \frac{R_1}{R_2}$	



F. FURTHER INVESTIGATION OF THE ALGORITHMS WITH NO DELAY FREE PATH IN PORT ONE, WITH THE ALGORITHMS WITH NO DELAY FREE PATH IN PORT TWO

A closer look to the two kinds of algorithms derived, i.e. algorithms with no delay free path from  $a_1$  to  $b_1$  and the algorithms with no delay free path from  $a_2$  to  $b_2$ , will reveal that both algorithms would behave very much like their equivalent LC structure matched to load or matched to the source as per the reciprocity theorem. Note that this can also be proven by writing the chain matrix equations

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} \mu_1 & \lambda_1 \\ \nu_1 & \kappa_1 \end{bmatrix} \begin{bmatrix} b_2 \\ a_2 \end{bmatrix} \quad \text{For no delay free path from } a_1 \text{ to } b_1$$

$$\text{or} \quad \begin{bmatrix} b_1 \\ a_1 \end{bmatrix} = \begin{bmatrix} \kappa_1 & \nu_1 \\ \lambda_1 & \mu_1 \end{bmatrix} \begin{bmatrix} a_2 \\ b_2 \end{bmatrix}$$

$$\text{or} \quad \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} = \begin{bmatrix} \kappa_1 & \nu_1 \\ \lambda_1 & \mu_1 \end{bmatrix}^{-1} \begin{bmatrix} b_1 \\ a_1 \end{bmatrix} \quad \text{With no delay free from } a_1 \text{ to } b_1$$

So if we interchange  $a_1$  and  $a_2$ , and also  $b_1$  and  $b_2$  we would get

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} \kappa_1 & \nu_1 \\ \lambda_1 & \mu_1 \end{bmatrix}^{-1} \begin{bmatrix} b_2 \\ a_2 \end{bmatrix} \quad \text{With no delay free from } a_2 \text{ to } b_2$$



Note that if we calculated the chain matrix for no delay free path from  $a_2$  to  $b_2$  i.e. the matrix

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} \mu_2 & \lambda_2 \\ \nu_2 & \kappa_2 \end{bmatrix} \begin{bmatrix} b_2 \\ a_2 \end{bmatrix}$$

Then the two matrices  $\begin{bmatrix} \kappa_1 & \nu_1 \\ \lambda_1 & \mu_1 \end{bmatrix}^{-1}$  and  $\begin{bmatrix} \mu_2 & \lambda_2 \\ \nu_2 & \kappa_2 \end{bmatrix}$

would be identical, i.e.

$$\begin{bmatrix} \frac{\mu_1}{\kappa_1 \mu_1 - \lambda_1 \nu_1} & \frac{-\nu_1}{\kappa_1 \mu_1 - \lambda_1 \nu_1} \\ \frac{-\lambda_1}{\kappa_1 \mu_1 - \lambda_1 \nu_1} & \frac{\kappa_1}{\kappa_1 \mu_1 - \lambda_1 \nu_1} \end{bmatrix} \equiv \begin{bmatrix} \mu_2 & \lambda_2 \\ \nu_2 & \kappa_2 \end{bmatrix}$$

Thus  $\mu_2 = \frac{\mu_1}{\kappa_1 \mu_1 - \lambda_1 \nu_1}$

$$\lambda_2 = \frac{\nu_1}{\lambda_1 \nu_1 - \kappa_1 \mu_1}$$

$$\nu_2 = \frac{\lambda_1}{\lambda_1 \nu_1 - \kappa_1 \mu_1}$$

$$\kappa_2 = \frac{\kappa_1}{\kappa_1 \mu_1 - \lambda_1 \nu_1}$$

## G. DESIGN EXAMPLE

As an application of the illustration of the techniques developed, a seventh order low pass Chebechev filter with the given specification in





Appendix 1A was designed and implemented in both time domain and frequency domain. As a comparison the same filter was designed using the conventional digital techniques. The computer program and output results for both cases are also given in the same appendix. As can be seen, they are completely identical in both time domain and frequency domain, thus assuring the validity of the theory developed in this chapter.

Note that in this example the double precision arithmetic was used, thus ensuring infinite precision for multiplier coefficients, to come up with exactly the same answer.

#### H. SUMMARY

In this chapter we developed the generalized algorithm for a two port wave digital filter step by step from the first principles of circuit theory and background given in Chapters II and III. The generalized LC system considered was an LC "T" Section. For a  $\pi$  section the procedure would have followed in the same way with the exactly similar end results.

The delay free path does play an essential role in the algorithms. Thus emphasis was given on this matter and whenever possible explanatory delay free paths were illustrated in the diagrams. To make things as clear as possible two worked examples are given, one with a simple section with no delay free path at the terminating source port and second example with a simple complex section with no delay free path at the terminating load port. Then we investigated the relationship between matched source algorithm with matched load algorithm of a given system. Finally a practical design of a wave digital filter using the already derived algorithms were implemented and the complete results and computer program are given in the Appendix 1B,-E.

Note that no reference is given at the end of this chapter. The only relevant references would be references [9], [11] of Chapter III.



## V. SENSITIVITY ANALYSIS OF THE WAVE DIGITAL FILTERS DUE TO TRUNCATION IN THE NUMBER OF BITS

### A. INTRODUCTION

The intent of this chapter is to investigate the behavior of wave digital filters to quantization in the number of bits in the multiplier coefficients. The wave digital filter has been credited in most of the published literature as being minimum sensitive to multiplier coefficient variations, and thus conjectured to be optimal in requiring the fewest number of significant places to achieve a given specification.

In this chapter, for a given analogue resistively terminated LC filter, three different wave digital filter algorithms are computed, and their behavior with respect to truncation in the number of bits of the multiplier coefficients are investigated. The results are compared with conventional digital filter design. It is important to note that for all the algorithms under investigation the bilinear transformation is used to make the comparison meaningful.

In order to avoid any confusion in this chapter we use the term "multiplier coefficient" for the coefficients of wave digital filter algorithms and the term "polynomial coefficient" to denote the coefficients  $a_6, a_5, \dots, a_0$  and  $b_6, b_5, \dots, b_0$  of  $H(s)$  or  $H(z)$ , of a given analogue or digital filter, respectively; where  $H(s)$  or  $H(z)$  are of the form

$$H(s) = \frac{1}{s^7 + a_6 s^6 + a_5 s^5 + a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0}$$
$$H(z) = \frac{K(z+1)^7}{z^7 + b_6 z^6 + b_5 z^5 + b_4 z^4 + b_3 z^3 + b_2 z^2 + b_1 z + b_0}$$



It is also important at this stage to note that since wave digital filter multipliers are derived directly from the original L's and C's, they can be thought of as a modified or matched L or C elements. Since wave digital multipliers are not derived from the polynomial coefficients of the transfer function of filter but from L's and C's of the analogue circuit, it is meaningful to compare the effect of bit quantization error on wave digital filter multipliers with the effect of bit quantization error on L's or C's in a conventional digital filter algorithm, and not to compare with the effect of bit quantization error on the filter polynomial coefficients. Based upon this argument, the original filter L's and C's are quantized in the conventional digital filter algorithms and used for comparison purposes.

#### B. COMPARATIVE STUDY OF THE ERROR DUE TO QUANTIZATION IN THE NUMBER OF BITS IN VARIOUS DIGITAL FILTER ALGORITHMS

Coefficient errors introduce perturbations in the zeros and poles of the transfer functions, which in turn manifest themselves as errors in the frequency response. In Chapter III Section B we defined general sensitivity, and in Section C we discussed parameter sensitivity, and in Section D sensitivity due to quantization in the number of bits. The comparison criteria we have used in this chapter is the root mean square error criteria given in equation (3.3) and repeated here in greater detail, i.e.

$$E_{\text{rms}} = \frac{1}{N+1} \sum_{i=0}^N [W(\omega_i) [\Delta H(\omega_i)]^2]^{\frac{1}{2}}$$

where

$W(\omega_i)$  is the frequency weighting function.

$\Delta H(\omega_i)$  is the difference in magnitude between infinite precision algorithm and the algorithm obtained in using a finite number of floating bits in the filter multiplier coefficients or component values.

N is the number of sampling points in the frequency domain.





$\omega_i$  is the frequency associated with the sampling point  $i$ . It should be pointed out that in floating point calculations, the mantissa of the floating number in the binary system is truncated to the specified length using sign magnitude representation. However the word length requirements of sign and exponent are not counted, since they do not enter into the calculations.

In total nine different normalized low pass seventh order filter algorithms were chosen for investigation from the handbook of filter synthesis [1]. These filters are

- i) seventh order .5 db ripple Chebyshev low pass filter with  $R_S=1$ ,  $R_L=1$
- ii) seventh order .1 db ripple Chebyshev low pass filter with  $R_S=1$ ,  $R_L=1$
- iii) seventh order Butterworth low pass filter with  $R_S=1$ ,  $R_L=1$
- iv) seventh order .5 db ripple Chebyshev low pass filter with  $R_S=0$ ,  $R_L=1$
- v) seventh order .1 db ripple Chebyshev low pass filter with  $R_S=0$ ,  $R_L=1$
- vi) seventh order Butterworth low pass filter with  $R_S=0$ ,  $R_L=1$
- vii) seventh order .5 db ripple Chebyshev low pass filter with  $R_S=10$ ,  $R_L=1$
- viii) seventh order .1 db ripple Chebyshev low pass filter with  $R_S=10$ ,  $R_L=1$
- ix) seventh order Butterworth low pass filter with  $R_S=10$ ,  $R_L=1$

Each filter was designed using five different algorithms.

- a) wave digital filter with simple sections, matched to the load (i.e. with no delay free path in port two)
- b) wave digital filter with complex sections, matched to the load (i.e. with no delay free path in port two)
- c) wave digital filter with complex sections but with reduced parameters and matched to the load (i.e. with no delay free path in port two)
- d) conventional direct digital filter of the form





$$H(z) = \frac{K(1+z^{-1})^7}{1+a_1z^{-1} + a_2z^{-2} + a_3z^{-3} + a_4z^{-4} + a_5z^{-5} + a_6z^{-6} + a_7z^{-7}}$$

(e) conventional cascaded digital filter of the form

$$H(z) = \frac{K(1+z^{-1})^7}{(1+\alpha_1z^{-1})(1+\alpha_2z^{-1} + \beta_2z^{-2})(1+\alpha_3z^{-1} + \beta_3z^{-2})(1+\alpha_4z^{-1} + \beta_4z^{-2})}$$

Note that all the filters are low pass and have zeros at  $z = 1$ . Thus there is no necessity for pole/zero pairing in order to optimize the response of the cascaded conventional digital filter. Thus all these examples are in a naturally optimum arrangement.

Double precision arithmetic used throughout in order to get practically the same result for all five algorithms with infinite precision arithmetic for the coefficient or component values.

The root mean square error in the frequency response for various filters under examination were found, and tabulated in Tables 5.1-9. The corresponding graphs are drawn in Figs. 5.1-9. Note that in order to avoid congestion the error graphs of conventional cascaded digital filter are not drawn.

A sample computer program for each of the five algorithms used is given in Appendix 2-A,F.

### C. STUDY OF THE RESULTS OBTAINED

The study of the graphs for all different algorithms clearly indicate the following points:



Filter No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.35652	.26039	.63499	.68992	.26039
2	.14008	.22310	.59928	.62552	.22310
3	.13521	.19962	.48117	.39121	.19962
4	.58828x10 <sup>-1</sup>	.95811x10 <sup>-1</sup>	.28575	.38759	.95811x10 <sup>-1</sup>
5	.29642x10 <sup>-1</sup>	.61900x10 <sup>-1</sup>	.14949	.23245	.61900x10 <sup>-1</sup>
6	.17023x10 <sup>-1</sup>	.3188 x10 <sup>-1</sup>	.61667x10 <sup>-1</sup>	.78677x10 <sup>-1</sup>	.31883x10 <sup>-1</sup>
7	.62789x10 <sup>-2</sup>	.12223x10 <sup>-1</sup>	.39411x10 <sup>-1</sup>	.46734x10 <sup>-1</sup>	.12223x10 <sup>-1</sup>
8	.43622x10 <sup>-2</sup>	.43154x10 <sup>-2</sup>	.19223x10 <sup>-1</sup>	.24077x10 <sup>-1</sup>	.43152x10 <sup>-2</sup>
9	.22449x10 <sup>-2</sup>	.24866x10 <sup>-2</sup>	.10759x10 <sup>-1</sup>	.12584x10 <sup>-1</sup>	.24859x10 <sup>-2</sup>
10	.10652x10 <sup>-2</sup>	.17172x10 <sup>-2</sup>	.49396x10 <sup>-2</sup>	.69758x10 <sup>-2</sup>	.17168x10 <sup>-2</sup>
11	.59664x10 <sup>-3</sup>	.82352x10 <sup>-3</sup>	.21325x10 <sup>-2</sup>	.11943x10 <sup>-2</sup>	.82280x10 <sup>-3</sup>
12	.35515x10 <sup>-3</sup>	.21312x10 <sup>-3</sup>	.91292x10 <sup>-3</sup>	.11787x10 <sup>-2</sup>	.21277x10 <sup>-3</sup>
13	.12313x10 <sup>-3</sup>	.14233x10 <sup>-3</sup>	.51889x10 <sup>-3</sup>	.67145x10 <sup>-3</sup>	.14237x10 <sup>-3</sup>
14	.37246x10 <sup>-4</sup>	.84985x10 <sup>-4</sup>	.16679x10 <sup>-3</sup>	.30656x10 <sup>-3</sup>	.85031x10 <sup>-4</sup>
15	.28579x10 <sup>-4</sup>	.39982x10 <sup>-4</sup>	.10920x10 <sup>-3</sup>	.15927x10 <sup>-3</sup>	.39202x10 <sup>-4</sup>
16	.15686x10 <sup>-4</sup>	.28431x10 <sup>-4</sup>	.85767x10 <sup>-4</sup>	.79318x10 <sup>-4</sup>	.28112x10 <sup>-4</sup>
17	.96697x10 <sup>-5</sup>	.13855x10 <sup>-4</sup>	.29565x10 <sup>-4</sup>	.65843x10 <sup>-4</sup>	.13557x10 <sup>-4</sup>
18	.56751x10 <sup>-5</sup>	.61544x10 <sup>-5</sup>	.15462x10 <sup>-4</sup>	.19280x10 <sup>-4</sup>	.61192x10 <sup>-5</sup>
19	.26435x10 <sup>-5</sup>	.12165x10 <sup>-5</sup>	.68256x10 <sup>-5</sup>	.82248x10 <sup>-5</sup>	.12755x10 <sup>-5</sup>
20	.12837x10 <sup>-5</sup>	.42522x10 <sup>-6</sup>	.34753x10 <sup>-5</sup>	.26740x10 <sup>-5</sup>	.65688x10 <sup>-6</sup>

Table 5.1. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .5 db ripple Chebyshev filter with  $R_s=1.0$ . (Case i). Note that all algorithms with infinite precision in the number of bits are identical.





Filter No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.14013	.24690	.65105	.70580	.24691
2	.12927	.12832	.56269	.64306	.12832
3	.91285x10 <sup>-1</sup>	.61380x10 <sup>-1</sup>	.47792	.44117	.61380x10 <sup>-1</sup>
4	.54932x10 <sup>-1</sup>	.29308x10 <sup>-1</sup>	.2885	.39337	.29308x10 <sup>-1</sup>
5	.34795x10 <sup>-1</sup>	.29308x10 <sup>-1</sup>	.12667	.20306	.29308x10 <sup>-1</sup>
6	.13549x10 <sup>-1</sup>	.2177 x10 <sup>-1</sup>	.64217x10 <sup>-1</sup>	.78850x10 <sup>-1</sup>	.47347
7	.51204x10 <sup>-2</sup>	.74590x10 <sup>-2</sup>	.26924x10 <sup>-1</sup>	.32705x10 <sup>-1</sup>	.74590x10 <sup>-2</sup>
8	.51204x10 <sup>-2</sup>	.24891x10 <sup>-2</sup>	.15956x10 <sup>-1</sup>	.14234x10 <sup>-1</sup>	.24892x10 <sup>-2</sup>
9	.23182x10 <sup>-2</sup>	.15261x10 <sup>-2</sup>	.87465x10 <sup>-2</sup>	.12626x10 <sup>-1</sup>	.15260x10 <sup>-2</sup>
10	.90195x10 <sup>-3</sup>	.10513x10 <sup>-2</sup>	.70939x10 <sup>-2</sup>	.85424x10 <sup>-2</sup>	.10513x10 <sup>-2</sup>
11	.35457x10 <sup>-3</sup>	.43836x10 <sup>-3</sup>	.24649x10 <sup>-2</sup>	.23713x10 <sup>-2</sup>	.43838x10 <sup>-3</sup>
12	.26058x10 <sup>-3</sup>	.32313x10 <sup>-3</sup>	.86570x10 <sup>-3</sup>	.11000x10 <sup>-2</sup>	.32304x10 <sup>-3</sup>
13	.12443x10 <sup>-3</sup>	.11146x10 <sup>-3</sup>	.24093x10 <sup>-3</sup>	.48572x10 <sup>-3</sup>	.11128x10 <sup>-3</sup>
14	.33816x10 <sup>-4</sup>	.49437x10 <sup>-4</sup>	.25903x10 <sup>-3</sup>	.55338x10 <sup>-3</sup>	.49478x10 <sup>-4</sup>
15	.22661x10 <sup>-4</sup>	.49437x10 <sup>-4</sup>	.16520x10 <sup>-3</sup>	.21469x10 <sup>-3</sup>	.49478x10 <sup>-4</sup>
16	.88987x10 <sup>-5</sup>	.21698x10 <sup>-4</sup>	.69406x10 <sup>-4</sup>	.11802x10 <sup>-3</sup>	.21610x10 <sup>-4</sup>
17	.58604x10 <sup>-5</sup>	.79930x10 <sup>-5</sup>	.20206x10 <sup>-4</sup>	.62724x10 <sup>-4</sup>	.79669x10 <sup>-5</sup>
18	.31205x10 <sup>-5</sup>	.40358x10 <sup>-5</sup>	.14222x10 <sup>-4</sup>	.27515x10 <sup>-4</sup>	.40597x10 <sup>-5</sup>
19	.69738x10 <sup>-6</sup>	.21025x10 <sup>-5</sup>	.62374x10 <sup>-5</sup>	.15394x10 <sup>-4</sup>	.23837x10 <sup>-5</sup>
20	.44562x10 <sup>-6</sup>	.90757x10 <sup>-6</sup>	.43746x10 <sup>-5</sup>	.92344x10 <sup>-5</sup>	.17713x10 <sup>-5</sup>

Table 5.2. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .1 db ripple Chebyshev filter with  $R_s=1.0$  (Case ii). Note that all algorithms with infinite precision in the number of Bits are identical.



Filter No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.15359	.37596	.62943	.70633	.55966
2	.98746x10 <sup>-1</sup>	.12358	.46779	.57919	.37285
3	.60823x10 <sup>-1</sup>	.68204x10 <sup>-1</sup>	.29004	.28707	.36805
4	.21522x10 <sup>-1</sup>	.24275x10 <sup>-1</sup>	.22520	.29007	.24275x10 <sup>-1</sup>
5	.14282x10 <sup>-1</sup>	.12006x10 <sup>-1</sup>	.13034	.18987	.12006x10 <sup>-1</sup>
6	.63848x10 <sup>-2</sup>	.12006x10 <sup>-1</sup>	.46460x10 <sup>-1</sup>	.80917x10 <sup>-1</sup>	.12006x10 <sup>-1</sup>
7	.63643x10 <sup>-2</sup>	.30730x10 <sup>-2</sup>	.23283x10 <sup>-1</sup>	.48606x10 <sup>-1</sup>	.30729x10 <sup>-2</sup>
8	.22526x10 <sup>-2</sup>	.58966x10 <sup>-3</sup>	.11585x10 <sup>-1</sup>	.17713x10 <sup>-1</sup>	.58954x10 <sup>-3</sup>
9	.75775x10 <sup>-3</sup>	.71222x10 <sup>-3</sup>	.49460x10 <sup>-2</sup>	.75595x10 <sup>-2</sup>	.71209x10 <sup>-3</sup>
10	.19093x10 <sup>-3</sup>	.35068x10 <sup>-3</sup>	.35341x10 <sup>-2</sup>	.39429x10 <sup>-2</sup>	.35063x10 <sup>-3</sup>
11	.14166x10 <sup>-3</sup>	.35068x10 <sup>-3</sup>	.18855x10 <sup>-2</sup>	.15097x10 <sup>-2</sup>	.35063x10 <sup>-3</sup>
12	.11492x10 <sup>-3</sup>	.16100x10 <sup>-3</sup>	.10525x10 <sup>-2</sup>	.13531x10 <sup>-2</sup>	.16087x10 <sup>-3</sup>
13	.30591x10 <sup>-4</sup>	.41379x10 <sup>-4</sup>	.42841x10 <sup>-3</sup>	.73249x10 <sup>-3</sup>	.41515x10 <sup>-4</sup>
14	.24121x10 <sup>-4</sup>	.13532x10 <sup>-4</sup>	.12317x10 <sup>-3</sup>	.38394x10 <sup>-3</sup>	.13593x10 <sup>-4</sup>
15	.69017x10 <sup>-5</sup>	.10017x10 <sup>-4</sup>	.11078x10 <sup>-3</sup>	.23523x10 <sup>-3</sup>	.99541x10 <sup>-5</sup>
16	.16907x10 <sup>-4</sup>	.53169x10 <sup>-5</sup>	.50313x10 <sup>-4</sup>	.99828x10 <sup>-4</sup>	.53993x10 <sup>-5</sup>
17	.12092x10 <sup>-5</sup>	.53169x10 <sup>-5</sup>	.22721x10 <sup>-4</sup>	.39670x10 <sup>-4</sup>	.53993x10 <sup>-5</sup>
18	.19942x10 <sup>-5</sup>	.12472x10 <sup>-5</sup>	.19869x10 <sup>-4</sup>	.25055x10 <sup>-4</sup>	.12047x10 <sup>-5</sup>
19	.12186x10 <sup>-5</sup>	.37570x10 <sup>-6</sup>	.90969x10 <sup>-5</sup>	.77472x10 <sup>-5</sup>	.68455x10 <sup>-6</sup>
20	.48111x10 <sup>-6</sup>	.43956x10 <sup>-6</sup>	.38317x10 <sup>-5</sup>	.74041x10 <sup>-5</sup>	.86858x10 <sup>-6</sup>

Table 5.3. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass Butterworth filter with  $R_s = 1.0$  (Case iii). Note that all algorithms with infinite precision in the number of bits are identical.





Filter No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.20668	.39184	.65288	.66916	.59279
2	.20668	.13616	.59238	.60740	.13616
3	.12086	.92858x10 <sup>-1</sup>	.47506	.40624	.92859x10 <sup>-1</sup>
4	.96439x10 <sup>-1</sup>	.40249x10 <sup>-1</sup>	.23035	.35495	.40249x10 <sup>-1</sup>
5	.37448x10 <sup>-1</sup>	.18858x10 <sup>-1</sup>	.10805	.20654	.18858x10 <sup>-1</sup>
6	.13055x10 <sup>-1</sup>	.92725x10 <sup>-2</sup>	.66144x10 <sup>-1</sup>	.77421x10 <sup>-1</sup>	.92719x10 <sup>-2</sup>
7	.72820x10 <sup>-2</sup>	.92725x10 <sup>-2</sup>	.41680x10 <sup>-1</sup>	.28951x10 <sup>-1</sup>	.92719x10 <sup>-2</sup>
8	.46487x10 <sup>-2</sup>	.47486x10 <sup>-2</sup>	.22673x10 <sup>-1</sup>	.26828x10 <sup>-1</sup>	.47485x10 <sup>-2</sup>
9	.26696x10 <sup>-2</sup>	.21632x10 <sup>-2</sup>	.10407x10 <sup>-1</sup>	.13094x10 <sup>-1</sup>	.21632x10 <sup>-2</sup>
10	.14931x10 <sup>-2</sup>	.11369x10 <sup>-2</sup>	.53214x10 <sup>-2</sup>	.69720x10 <sup>-2</sup>	.11365x10 <sup>-2</sup>
11	.52650x10 <sup>-3</sup>	.40809x10 <sup>-3</sup>	.30710x10 <sup>-2</sup>	.11811x10 <sup>-2</sup>	.40696x10 <sup>-3</sup>
12	.22758x10 <sup>-3</sup>	.15602x10 <sup>-3</sup>	.18877x10 <sup>-2</sup>	.12389x10 <sup>-2</sup>	.15576x10 <sup>-3</sup>
13	.12816x10 <sup>-3</sup>	.87660x10 <sup>-4</sup>	.89403x10 <sup>-3</sup>	.41425x10 <sup>-3</sup>	.87011x10 <sup>-4</sup>
14	.66198x10 <sup>-4</sup>	.89247x10 <sup>-4</sup>	.30364x10 <sup>-3</sup>	.38149x10 <sup>-3</sup>	.88559x10 <sup>-4</sup>
15	.31131x10 <sup>-4</sup>	.37404x10 <sup>-4</sup>	.11601x10 <sup>-3</sup>	.19776x10 <sup>-3</sup>	.37172x10 <sup>-4</sup>
16	.22886x10 <sup>-4</sup>	.13096x10 <sup>-4</sup>	.43989x10 <sup>-4</sup>	.89407x10 <sup>-4</sup>	.13564x10 <sup>-4</sup>
17	.75737x10 <sup>-5</sup>	.94055x10 <sup>-5</sup>	.30817x10 <sup>-4</sup>	.40262x10 <sup>-4</sup>	.91978x10 <sup>-5</sup>
18	.31208x10 <sup>-5</sup>	.5592 x10 <sup>-5</sup>	.23248x10 <sup>-4</sup>	.19890x10 <sup>-4</sup>	.55810x10 <sup>-5</sup>
19	.20293x10 <sup>-5</sup>	.19806x10 <sup>-5</sup>	.93558x10 <sup>-5</sup>	.60506x10 <sup>-5</sup>	.26475x10 <sup>-5</sup>
20	.12448x10 <sup>-5</sup>	.81813x10 <sup>-6</sup>	.49603x10 <sup>-5</sup>	.54628x10 <sup>-5</sup>	.11476x10 <sup>-5</sup>

Table 5.4. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .5 db ripple Chebyshev filter with  $R_s=0$  (Case iv). Note that all algorithms with infinite precision in the number of bits are identical.



Filter No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.25667	.38205	.67249	.68073	.58086
2	.18028	.31179	.58285	.61156	.31179
3	.11517	.15140	.46909	.39879	.23871
4	.46371x10 <sup>-1</sup>	.49949x10 <sup>-1</sup>	.22673	.36262	.37667
5	.26852x10 <sup>-1</sup>	.15935x10 <sup>-1</sup>	.15270	.19724	.15935x10 <sup>-1</sup>
6	.10936x10 <sup>-1</sup>	.88682x10 <sup>-2</sup>	.70773x10 <sup>-1</sup>	.11305	.44566
7	.41706x10 <sup>-2</sup>	.66298x10 <sup>-2</sup>	.37405x10 <sup>-1</sup>	.75338x10 <sup>-1</sup>	.66297x10 <sup>-2</sup>
8	.22884x10 <sup>-2</sup>	.27692x10 <sup>-2</sup>	.21136x10 <sup>-1</sup>	.35243x10 <sup>-1</sup>	.27690x10 <sup>-2</sup>
9	.19746x10 <sup>-2</sup>	.16209x10 <sup>-2</sup>	.75015x10 <sup>-1</sup>	.20651x10 <sup>-1</sup>	.16208x10 <sup>-2</sup>
10	.89626x10 <sup>-3</sup>	.96046x10 <sup>-3</sup>	.45627x10 <sup>-2</sup>	.91342x10 <sup>-2</sup>	.96083x10 <sup>-3</sup>
11	.39825x10 <sup>-3</sup>	.38064x10 <sup>-3</sup>	.2725 x10 <sup>-2</sup>	.32694x10 <sup>-2</sup>	.38038x10 <sup>-3</sup>
12	.10908x10 <sup>-3</sup>	.24080x10 <sup>-3</sup>	.10593x10 <sup>-2</sup>	.34804x10 <sup>-3</sup>	.24081x10 <sup>-3</sup>
13	.93393x10 <sup>-4</sup>	.88964x10 <sup>-4</sup>	.67468x10 <sup>-3</sup>	.34342x10 <sup>-3</sup>	.89049x10 <sup>-4</sup>
14	.39722x10 <sup>-4</sup>	.35992x10 <sup>-4</sup>	.3080 x10 <sup>-3</sup>	.31101x10 <sup>-3</sup>	.35926x10 <sup>-4</sup>
15	.25247x10 <sup>-4</sup>	.31273x10 <sup>-4</sup>	.17009x10 <sup>-3</sup>	.28543x10 <sup>-3</sup>	.31234x10 <sup>-4</sup>
16	.20389x10 <sup>-4</sup>	.11650x10 <sup>-4</sup>	.83274x10 <sup>-4</sup>	.10079x10 <sup>-3</sup>	.11775x10 <sup>-4</sup>
17	.90891x10 <sup>-5</sup>	.63657x10 <sup>-5</sup>	.34725x10 <sup>-4</sup>	.60767x10 <sup>-4</sup>	.67364x10 <sup>-5</sup>
18	.37639x10 <sup>-5</sup>	.42258x10 <sup>-5</sup>	.21278x10 <sup>-4</sup>	.17197x10 <sup>-4</sup>	.41910x10 <sup>-5</sup>
19	.15389x10 <sup>-5</sup>	.23810x10 <sup>-5</sup>	.11086x10 <sup>-4</sup>	.19679x10 <sup>-5</sup>	.22792x10 <sup>-5</sup>
20	.92275x10 <sup>-6</sup>	.12793x10 <sup>-5</sup>	.32788x10 <sup>-5</sup>	.15601x10 <sup>-5</sup>	.13736x10 <sup>-5</sup>

Table 5.5. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .1 db ripple Chebyshev filter with  $R_s=0$  (Case v). Note that all algorithms with infinite precision in the number of bits are identical.





Filter No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	.21385	.35023	.63007	.67050	.60960
2	.21684	.22495	.34968	.48535	.22496
3	.11543	.10316	.26403	.31619	.45125
4	.37482x10 <sup>-1</sup>	.21924x10 <sup>-1</sup>	.21129	.24584	.50695
5	.15989x10 <sup>-1</sup>	.21324x10 <sup>-1</sup>	.10929	.70189x10 <sup>-1</sup>	.13494
6	.65223x10 <sup>-2</sup>	.11353x10 <sup>-1</sup>	.57963x10 <sup>-1</sup>	.64994x10 <sup>-1</sup>	.11353x10 <sup>-1</sup>
7	.65223x10 <sup>-2</sup>	.27292x10 <sup>-2</sup>	.32092x10 <sup>-1</sup>	.24383x10 <sup>-1</sup>	.41748
8	.24615x10 <sup>-2</sup>	.25787x10 <sup>-2</sup>	.20078x10 <sup>-1</sup>	.24989x10 <sup>-1</sup>	.33654
9	.65170x10 <sup>-3</sup>	.16855x10 <sup>-2</sup>	.91341x10 <sup>-2</sup>	.10063x10 <sup>-1</sup>	.16854x10 <sup>-2</sup>
10	.41747x10 <sup>-3</sup>	.95453x10 <sup>-3</sup>	.56683x10 <sup>-2</sup>	.14316x10 <sup>-2</sup>	.95454x10 <sup>-3</sup>
11	.10700x10 <sup>-3</sup>	.46437x10 <sup>-3</sup>	.23622x10 <sup>-2</sup>	.90073x10 <sup>-3</sup>	.46429x10 <sup>-3</sup>
12	.21860x10 <sup>-3</sup>	.17166x10 <sup>-3</sup>	.76595x10 <sup>-3</sup>	.67584x10 <sup>-3</sup>	.17162x10 <sup>-3</sup>
13	.13791x10 <sup>-3</sup>	.59632x10 <sup>-4</sup>	.38607x10 <sup>-3</sup>	.67584x10 <sup>-3</sup>	.59672x10 <sup>-4</sup>
14	.73024x10 <sup>-4</sup>	.38676x10 <sup>-4</sup>	.13104x10 <sup>-3</sup>	.29072x10 <sup>-3</sup>	.38583x10 <sup>-4</sup>
15	.22530x10 <sup>-4</sup>	.24061x10 <sup>-4</sup>	.64822x10 <sup>-4</sup>	.10959x10 <sup>-3</sup>	.24105x10 <sup>-4</sup>
16	.12791x10 <sup>-4</sup>	.89135x10 <sup>-5</sup>	.37915x10 <sup>-4</sup>	.19217x10 <sup>-4</sup>	.88321x10 <sup>-5</sup>
17	.54724x10 <sup>-5</sup>	.31105x10 <sup>-5</sup>	.17687x10 <sup>-4</sup>	.15600x10 <sup>-4</sup>	.33782x10 <sup>-5</sup>
18	.10315x10 <sup>-5</sup>	.83639x10 <sup>-6</sup>	.13154x10 <sup>-4</sup>	.17382x10 <sup>-4</sup>	.10057x10 <sup>-5</sup>
19	.11843x10 <sup>-5</sup>	.94314x10 <sup>-6</sup>	.17485x10 <sup>-4</sup>	.57124x10 <sup>-5</sup>	.18559x10 <sup>-5</sup>
20	.56671x10 <sup>-6</sup>	.61413x10 <sup>-6</sup>	.42001x10 <sup>-5</sup>	.30040x10 <sup>-5</sup>	.64950x10 <sup>-6</sup>

Table 5.6. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass Butterworth filter with  $R_s=0.0$  (Case vi). Note that all algorithms with infinite precision in the number of bits are identical.



Filter No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	3.4045	.36760	.39479	.64376	.36760
2	2.5171	.25718	.25614	.44176	.25718
3	.75203	.12198	.20941	.17782	.12198
4	.10343	.59672x10 <sup>-1</sup>	.20967	.32729	.59673x10 <sup>-1</sup>
5	.71300x10 <sup>-1</sup>	.58777x10 <sup>-1</sup>	.91164x10 <sup>-1</sup>	.16175	.58778x10 <sup>-1</sup>
6	.63749x10 <sup>-1</sup>	.22581x10 <sup>-1</sup>	.23961x10 <sup>-1</sup>	.75455x10 <sup>-1</sup>	.22582x10 <sup>-1</sup>
7	.52302x10 <sup>-1</sup>	.10077x10 <sup>-1</sup>	.15663x10 <sup>-1</sup>	.26816x10 <sup>-1</sup>	.10078x10 <sup>-1</sup>
8	.10162x10 <sup>-1</sup>	.37212x10 <sup>-2</sup>	.72677x10 <sup>-2</sup>	.31405x10 <sup>-2</sup>	.37218x10 <sup>-2</sup>
9	.65050x10 <sup>-2</sup>	.12500x10 <sup>-2</sup>	.39261x10 <sup>-2</sup>	.24761x10 <sup>-2</sup>	.12502x10 <sup>-2</sup>
10	.54846x10 <sup>-2</sup>	.75861x10 <sup>-3</sup>	.24435x10 <sup>-2</sup>	.25190x10 <sup>-2</sup>	.75847x10 <sup>-3</sup>
11	.51118x10 <sup>-2</sup>	.50342x10 <sup>-3</sup>	.32102x10 <sup>-2</sup>	.35433x10 <sup>-2</sup>	.50396x10 <sup>-3</sup>
12	.23803x10 <sup>-2</sup>	.43317x10 <sup>-3</sup>	.16176x10 <sup>-2</sup>	.1225 x10 <sup>-2</sup>	.43352x10 <sup>-3</sup>
13	.77886x10 <sup>-3</sup>	.18262x10 <sup>-3</sup>	.38714x10 <sup>-3</sup>	.39487x10 <sup>-3</sup>	.18315x10 <sup>-3</sup>
14	.72844x10 <sup>-4</sup>	.12202x10 <sup>-3</sup>	.27273x10 <sup>-3</sup>	.43587x10 <sup>-3</sup>	.12213x10 <sup>-3</sup>
15	.95398x10 <sup>-4</sup>	.52849x10 <sup>-4</sup>	.90359x10 <sup>-4</sup>	.80415x10 <sup>-4</sup>	.53353x10 <sup>-4</sup>
16	.68363x10 <sup>-4</sup>	.33381x10 <sup>-4</sup>	.39191x10 <sup>-4</sup>	.72033x10 <sup>-4</sup>	.33771x10 <sup>-4</sup>
17	.80049x10 <sup>-4</sup>	.13892x10 <sup>-4</sup>	.51176x10 <sup>-4</sup>	.36564x10 <sup>-4</sup>	.13793x10 <sup>-4</sup>
18	.34872x10 <sup>-4</sup>	.52833x10 <sup>-5</sup>	.18834x10 <sup>-4</sup>	.19394x10 <sup>-4</sup>	.54028x10 <sup>-5</sup>
19	.11083x10 <sup>-4</sup>	.16061x10 <sup>-5</sup>	.28221x10 <sup>-5</sup>	.93569x10 <sup>-5</sup>	.16240x10 <sup>-5</sup>
20	.93252x10 <sup>-5</sup>	.11946x10 <sup>-5</sup>	.47440x10 <sup>-5</sup>	.54222x10 <sup>-5</sup>	.12857x10 <sup>-5</sup>

Table 5.7. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .5 db ripple Chebyshev filter with  $R=10$  (Case vii). Note that all algorithms with in-finite precision in the number of bits are identical.





Filter No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	4.9330	.34028	.22253	.60537	.34028
2	2.3234	.25423	.25134	.39941	.25423
3	.60260	.11162	.24523	.19258	.11162
4	.28045x10 <sup>-1</sup>	.11162	.25260	.34407	.11162
5	.19225x10 <sup>-1</sup>	.51472x10 <sup>-1</sup>	.81241x10 <sup>-1</sup>	.14301	.51472x10 <sup>-1</sup>
6	.16671x10 <sup>-1</sup>	.17350x10 <sup>-1</sup>	.41152x10 <sup>-1</sup>	.52584x10 <sup>-1</sup>	.1735 x10 <sup>-1</sup>
7	.14588x10 <sup>-1</sup>	.90435x10 <sup>-2</sup>	.53493x10 <sup>-2</sup>	.41214x10 <sup>-1</sup>	.90437x10 <sup>-2</sup>
8	.14292x10 <sup>-1</sup>	.50883x10 <sup>-2</sup>	.46919x10 <sup>-2</sup>	.12899x10 <sup>-1</sup>	.50883x10 <sup>-2</sup>
9	.14518x10 <sup>-1</sup>	.18920x10 <sup>-2</sup>	.99608x10 <sup>-2</sup>	.82507x10 <sup>-2</sup>	.18920x10 <sup>-2</sup>
10	.34443x10 <sup>-2</sup>	.50810x10 <sup>-3</sup>	.86433x10 <sup>-3</sup>	.36649x10 <sup>-2</sup>	.50805x10 <sup>-3</sup>
11	.32681x10 <sup>-2</sup>	.48189x10 <sup>-3</sup>	.15036x10 <sup>-2</sup>	.23079x10 <sup>-2</sup>	.48178x10 <sup>-3</sup>
12	.51894x10 <sup>-3</sup>	.27238x10 <sup>-3</sup>	.12471x10 <sup>-2</sup>	.21315x10 <sup>-3</sup>	.27251x10 <sup>-3</sup>
13	.65150x10 <sup>-3</sup>	.14401x10 <sup>-3</sup>	.28402x10 <sup>-3</sup>	.81029x10 <sup>-3</sup>	.14397x10 <sup>-3</sup>
14	.69939x10 <sup>-3</sup>	.86010x10 <sup>-4</sup>	.46751x10 <sup>-3</sup>	.32312x10 <sup>-3</sup>	.86135x10 <sup>-4</sup>
15	.33369x10 <sup>-3</sup>	.20719x10 <sup>-4</sup>	.21478x10 <sup>-3</sup>	.15872x10 <sup>-3</sup>	.20736x10 <sup>-4</sup>
16	.15398x10 <sup>-3</sup>	.14006x10 <sup>-4</sup>	.78033x10 <sup>-4</sup>	.79171x10 <sup>-4</sup>	.14442x10 <sup>-2</sup>
17	.72318x10 <sup>-4</sup>	.80534x10 <sup>-5</sup>	.38563x10 <sup>-4</sup>	.46419x10 <sup>-4</sup>	.84066x10 <sup>-5</sup>
18	.35157x10 <sup>-4</sup>	.23806x10 <sup>-5</sup>	.23438x10 <sup>-4</sup>	.17891x10 <sup>-4</sup>	.24410x10 <sup>-5</sup>
19	.14323x10 <sup>-4</sup>	.13523x10 <sup>-5</sup>	.87165x10 <sup>-5</sup>	.91258x10 <sup>-5</sup>	.15303x10 <sup>-5</sup>
20	.39038x10 <sup>-5</sup>	.63022x10 <sup>-6</sup>	.11030x10 <sup>-5</sup>	.50819x10 <sup>-5</sup>	.10907x10 <sup>-5</sup>

Table 5.8. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass .1 db ripple Chebyshev filter with  $R_s=10$  (Case viii). Note that all algorithms with infinite precision in the number of bits are identical.



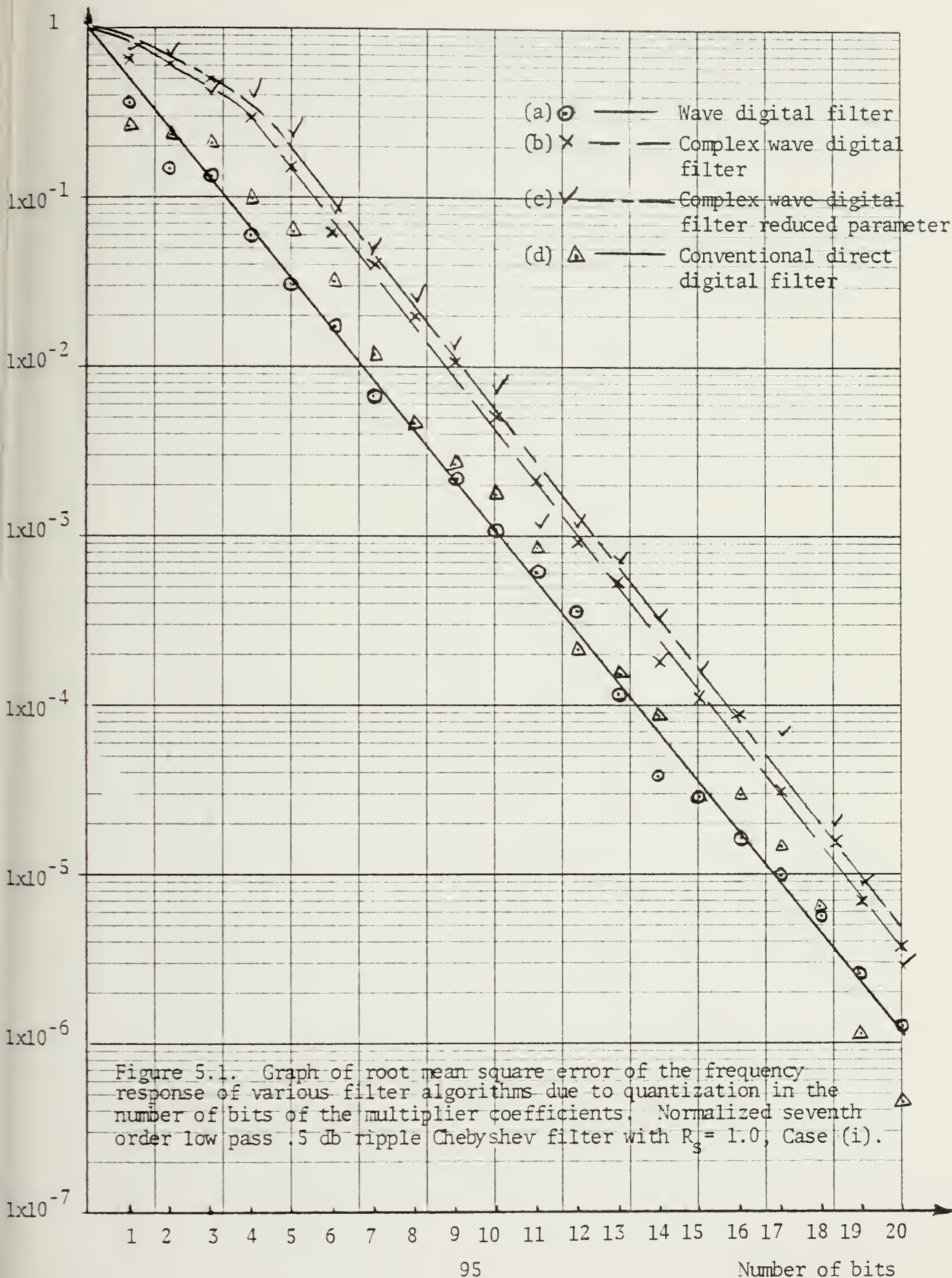
Filter No. of Bits	Simple Wave Digital Filter	Conventional Direct Digital Filter	Complex Wave Digital Filter	Complex Wave Digital Filter Reduced Parameter	Conventional Cascaded Digital Filter
1	4.9388	.31595	.43082	.42707	.31595
2	1.7731	.19920	.21663	.20286	.19920
3	.73610	.53069x10 <sup>-1</sup>	.21097	.27391	.33561
4	.19259x10 <sup>-1</sup>	.40313x10 <sup>-1</sup>	.18841	.20363	.32817
5	.19821x10 <sup>-1</sup>	.19975x10 <sup>-1</sup>	.91724x10 <sup>-1</sup>	.13022	.51223
6	.17707x10 <sup>-1</sup>	.12376x10 <sup>-1</sup>	.26432x10 <sup>-1</sup>	.58937x10 <sup>-1</sup>	.29823
7	.96715x10 <sup>-2</sup>	.30099x10 <sup>-2</sup>	.13196x10 <sup>-1</sup>	.18244x10 <sup>-1</sup>	.32534
8	.12042x10 <sup>-1</sup>	.29839x10 <sup>-2</sup>	.27875x10 <sup>-2</sup>	.13429x10 <sup>-1</sup>	.29839x10 <sup>-2</sup>
9	.11451x10 <sup>-1</sup>	.12108x10 <sup>-2</sup>	.57712x10 <sup>-2</sup>	.60203x10 <sup>-2</sup>	.12108x10 <sup>-2</sup>
10	.19437x10 <sup>-2</sup>	.45825x10 <sup>-3</sup>	.17493x10 <sup>-2</sup>	.40077x10 <sup>-2</sup>	.45820x10 <sup>-3</sup>
11	.12416x10 <sup>-2</sup>	.45719x10 <sup>-3</sup>	.52506x10 <sup>-3</sup>	.17260x10 <sup>-2</sup>	.45717x10 <sup>-3</sup>
12	.14031x10 <sup>-2</sup>	.18987x10 <sup>-3</sup>	.72945x10 <sup>-3</sup>	.75322x10 <sup>-3</sup>	.18979x10 <sup>-3</sup>
13	.22524x10 <sup>-3</sup>	.72993x10 <sup>-4</sup>	.20080x10 <sup>-3</sup>	.19607x10 <sup>-3</sup>	.72942x10 <sup>-4</sup>
14	.16257x10 <sup>-3</sup>	.36324x10 <sup>-4</sup>	.87813x10 <sup>-4</sup>	.96061x10 <sup>-4</sup>	.36521x10 <sup>-4</sup>
15	.18233x10 <sup>-3</sup>	.15476x10 <sup>-4</sup>	.10840x10 <sup>-3</sup>	.86385x10 <sup>-4</sup>	.15527x10 <sup>-4</sup>
16	.11824x10 <sup>-4</sup>	.10164x10 <sup>-4</sup>	.29437x10 <sup>-4</sup>	.53998x10 <sup>-4</sup>	.10109x10 <sup>-4</sup>
17	.11995x10 <sup>-4</sup>	.59171x10 <sup>-5</sup>	.11695x10 <sup>-4</sup>	.33243x10 <sup>-4</sup>	.61673x10 <sup>-5</sup>
18	.12213x10 <sup>-4</sup>	.36245x10 <sup>-5</sup>	.55384x10 <sup>-5</sup>	.13669x10 <sup>-4</sup>	.37382x10 <sup>-5</sup>
19	.14193x10 <sup>-4</sup>	.14071x10 <sup>-5</sup>	.92956x10 <sup>-5</sup>	.72502x10 <sup>-5</sup>	.15625x10 <sup>-5</sup>
20	.32546x10 <sup>-5</sup>	.38308x10 <sup>-6</sup>	.12126x10 <sup>-5</sup>	.32347x10 <sup>-5</sup>	.39579x10 <sup>-6</sup>

Table 5.9. Root mean square error of the frequency response of various filter algorithms due to the quantization in number of bits of the multipliers for the normalized 7th order low pass Butterworth filter with  $R_c=10$  (Case ix). Note that all algorithms with infinite precision in the number of bits are identical.





# Root mean square error





# Root mean square error

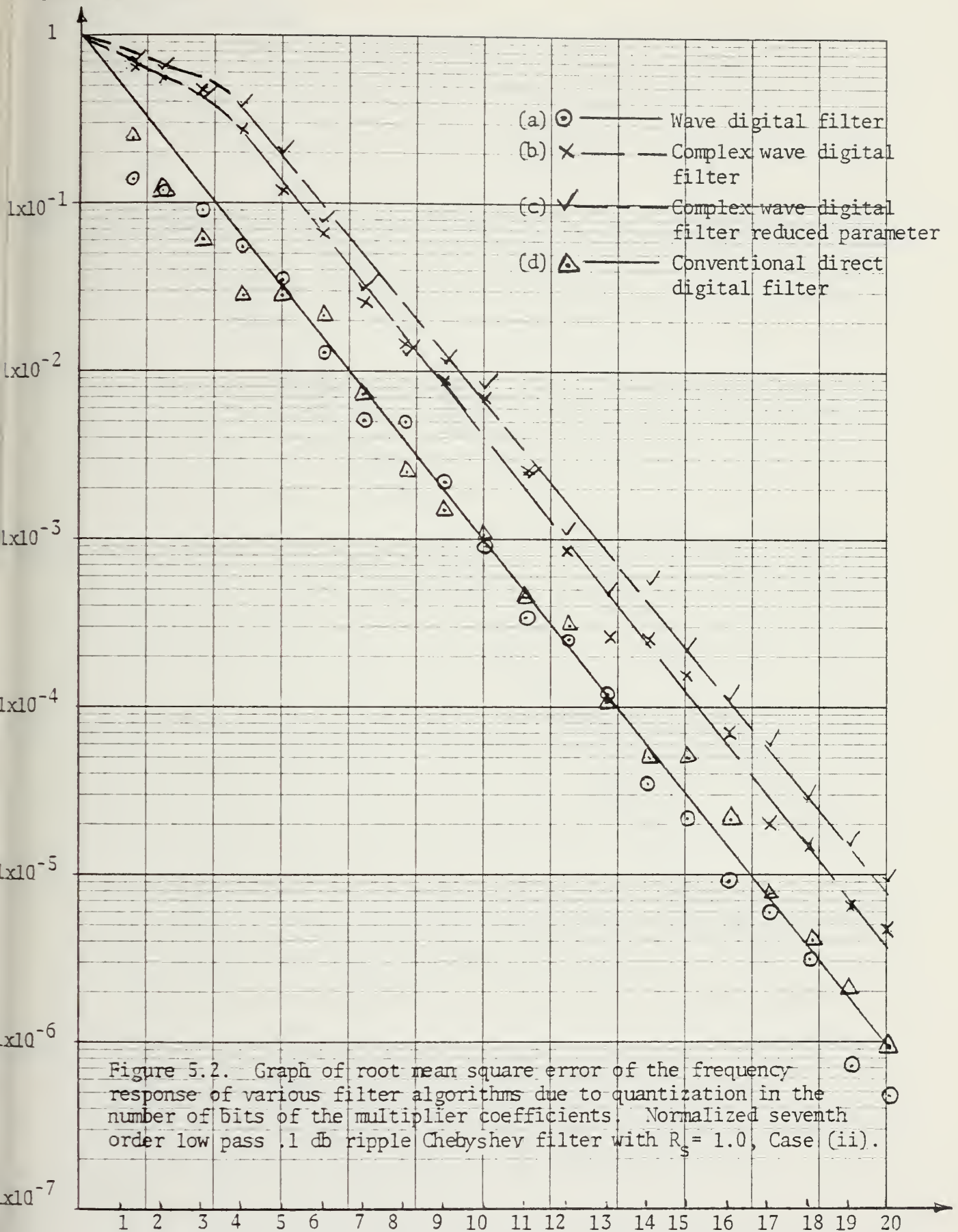
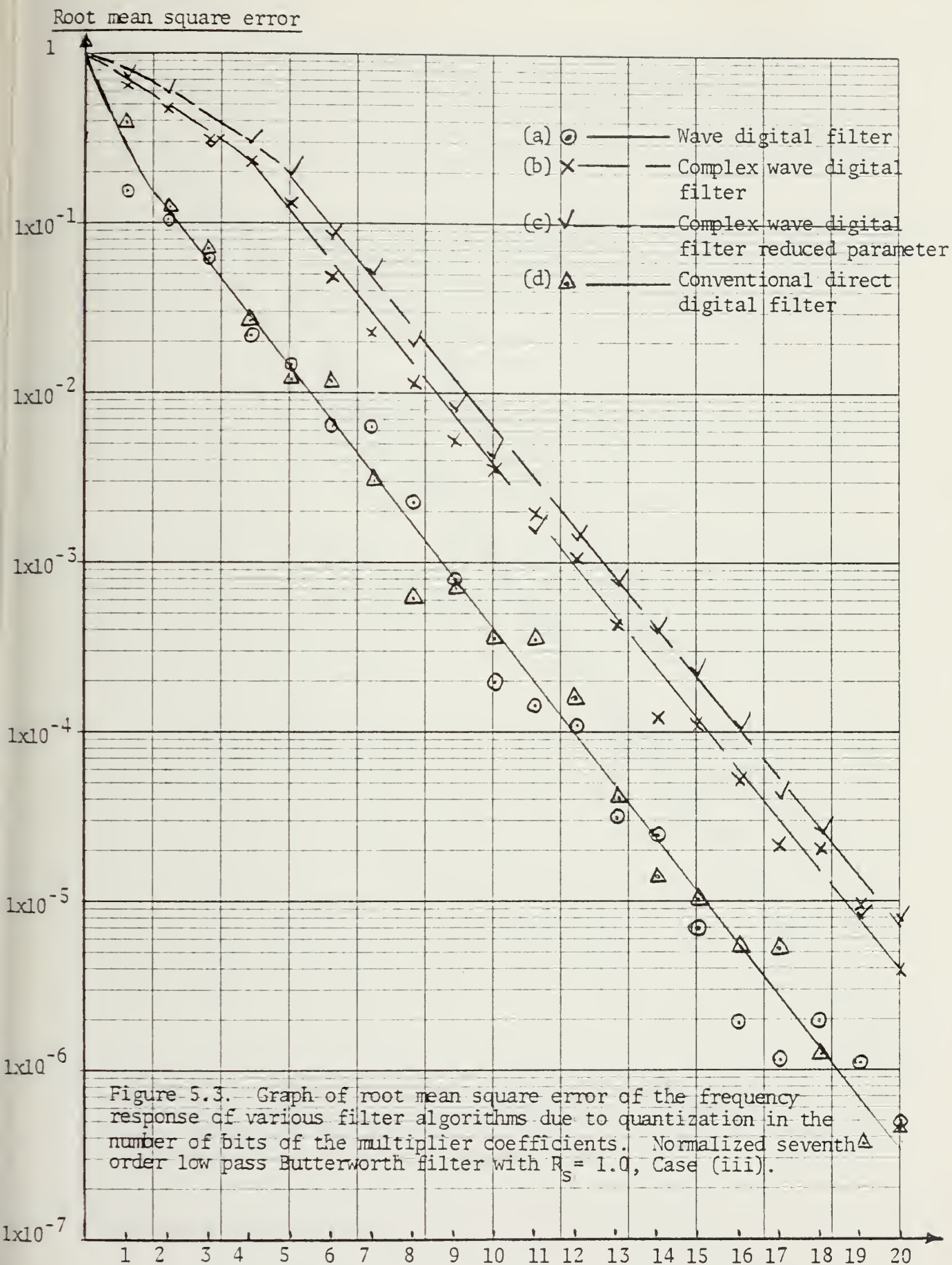


Figure 5.2. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass .1 db ripple Chebyshev filter with  $R_s = 1.0$ , Case (ii).

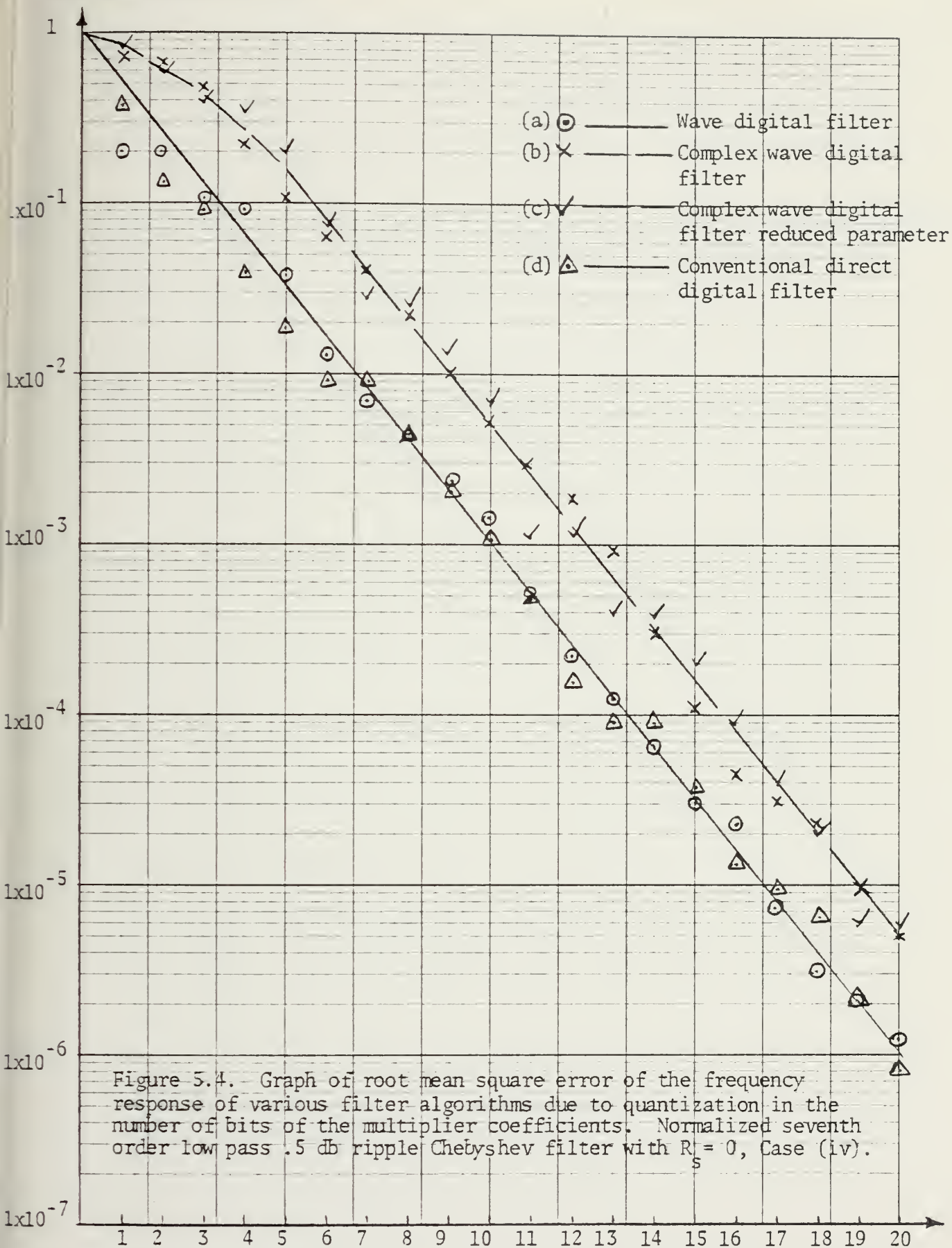








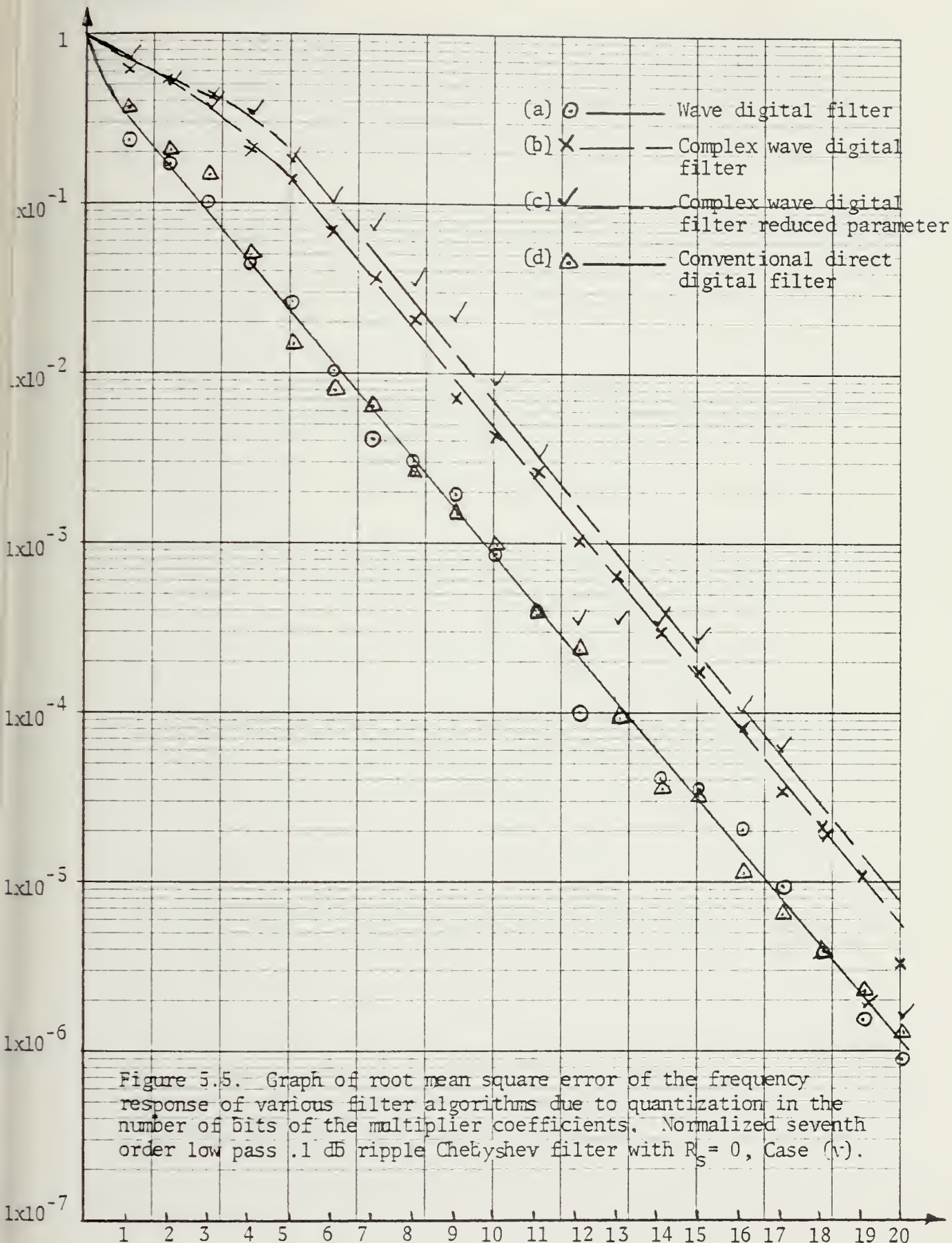
# Root mean square error



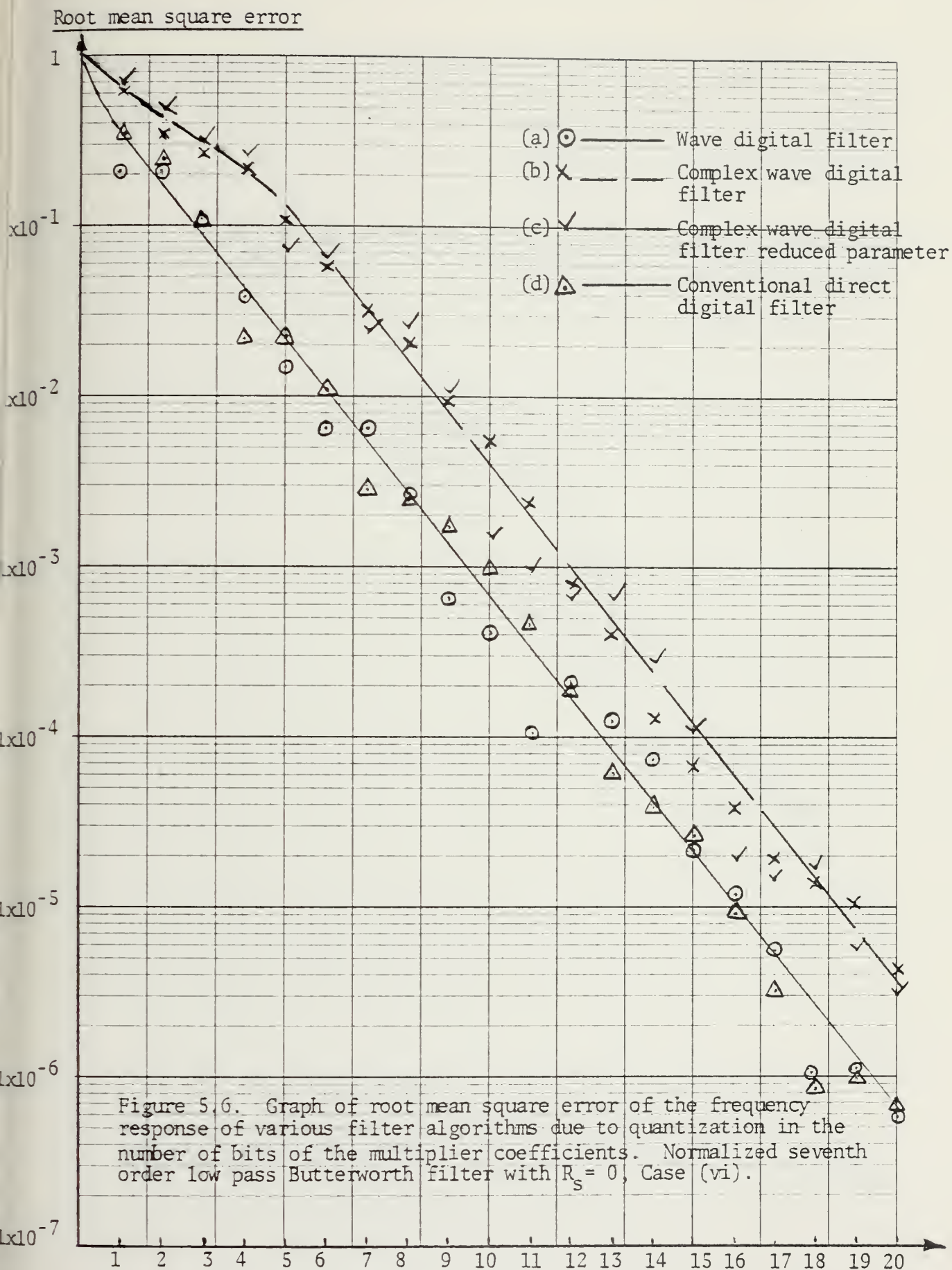




# Root mean square error

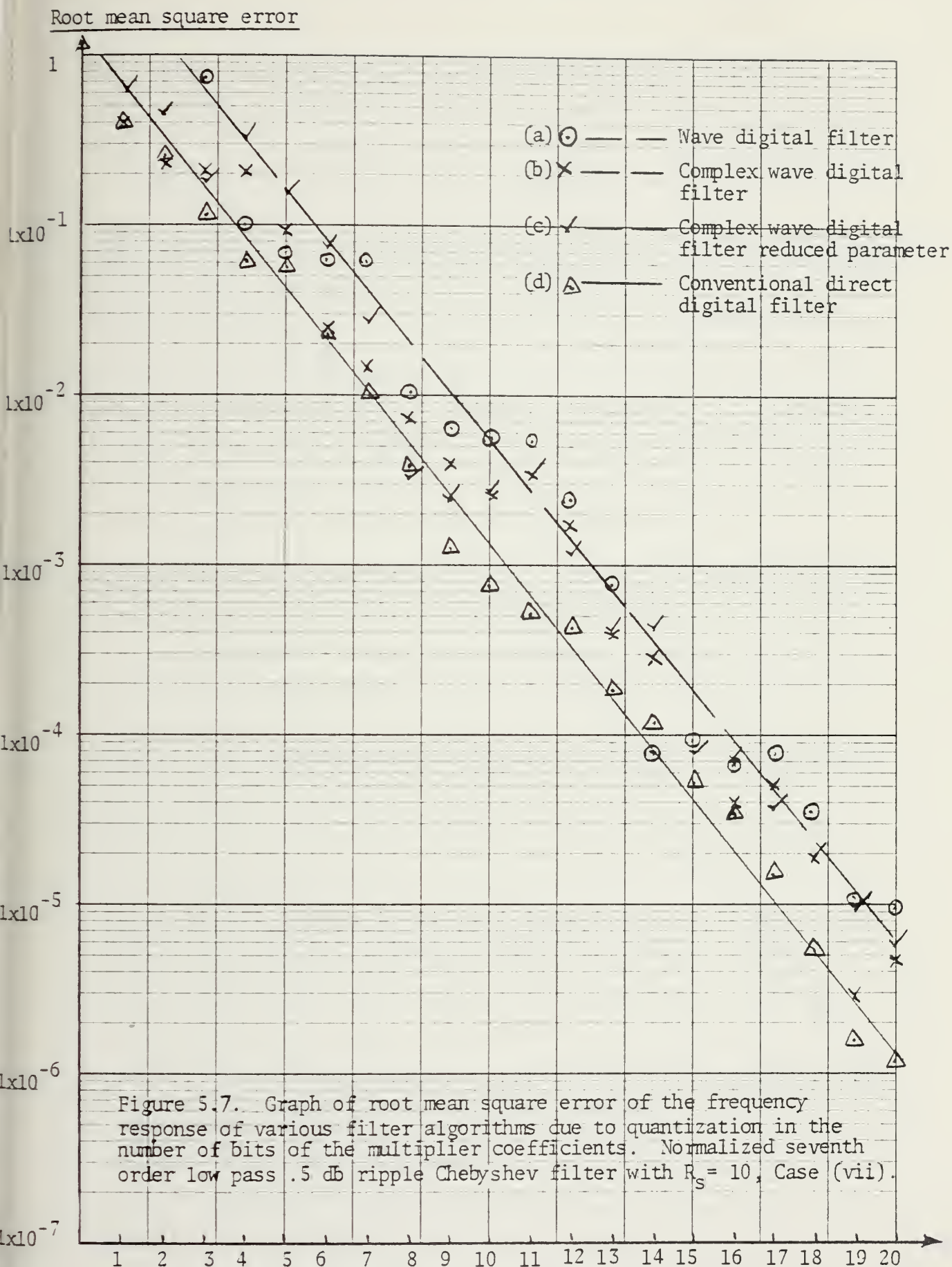






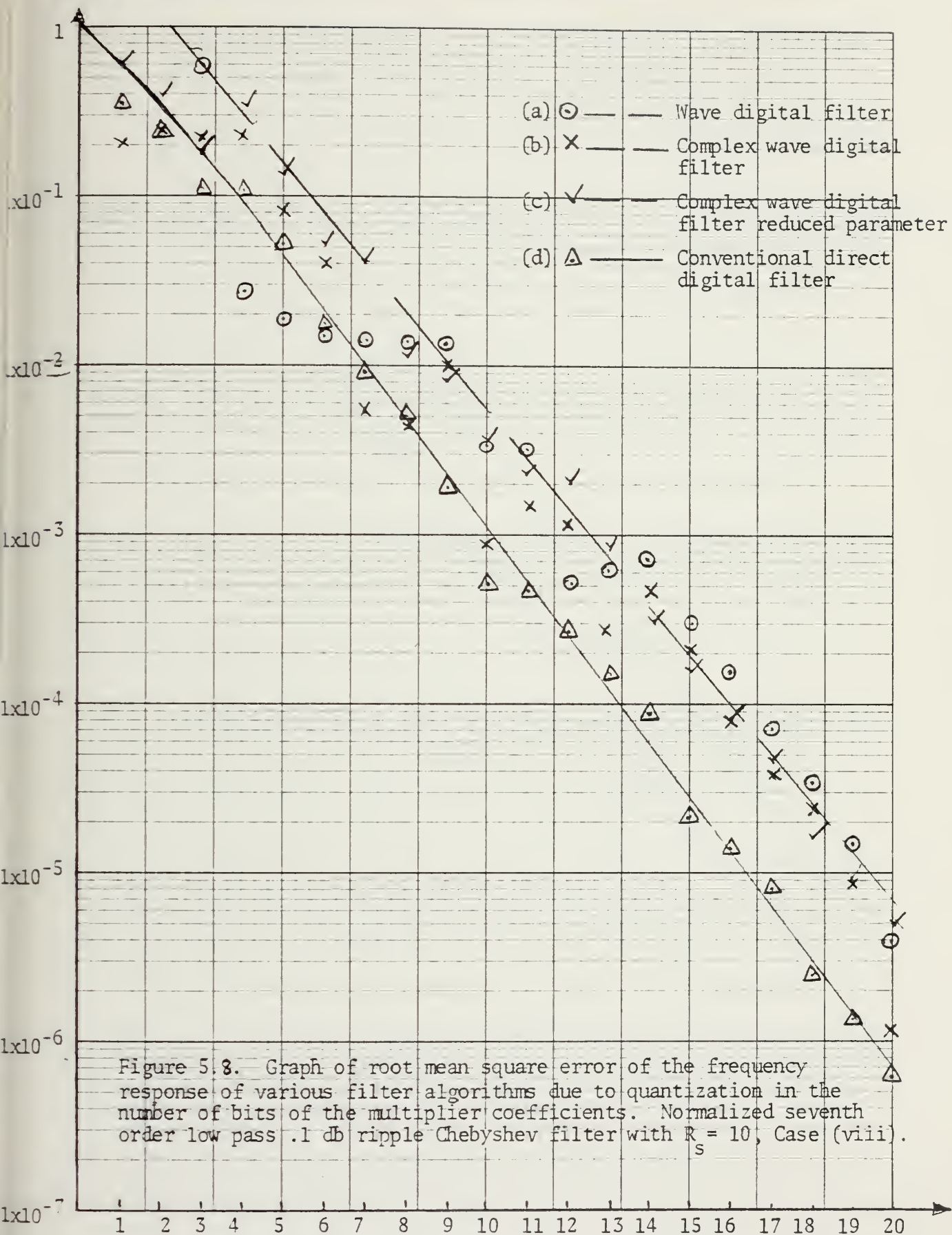








# Root mean square error







# Root mean square error

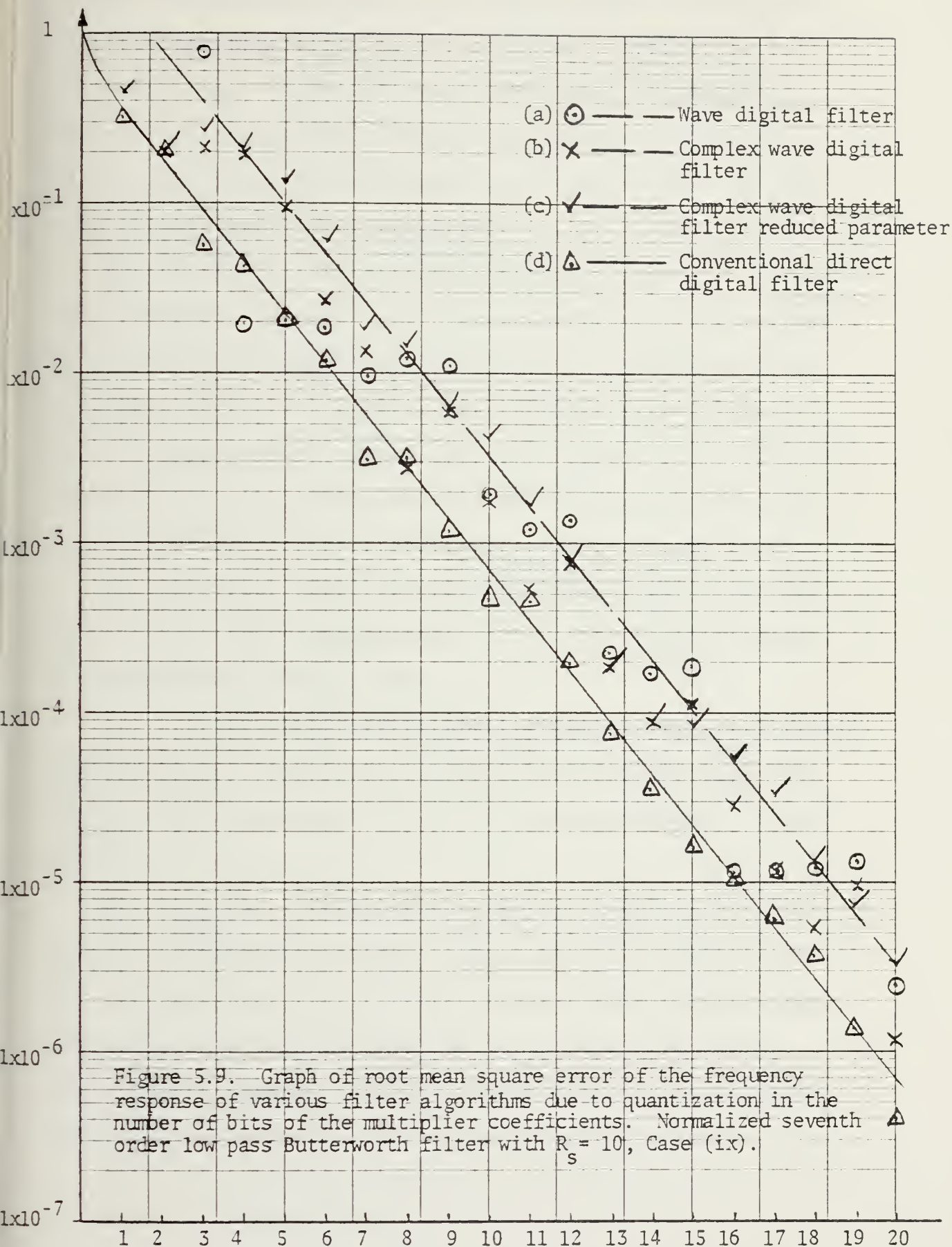


Figure 5.9. Graph of root mean square error of the frequency response of various filter algorithms due to quantization in the number of bits of the multiplier coefficients. Normalized seventh order low pass Butterworth filter with  $R_s = 10$ , Case (ix).



(1) The logarithm of rms error versus the number of bits for the direct conventional digital algorithms follows a straight line for all seventh order filters and they all pass close to the point of error = 1 at bits = 0 with a slope of approximately 3 db per bit. Also the rms error of the conventional direct digital filter is never worse than the other filter algorithms.

(2) For filters with source resistance  $R_s = 1$  and  $R_s = 0$ , the rms error graphs for the simple wave digital filters are exactly the same as that of direct conventional digital filter, with the same slope. For filters with  $R_s = 10$ , the rms error graph for the wave digital filter is worse than the conventional direct digital filter. This error behavior is very interesting and shows that wave digital filters do have a higher sensitivity to filter terminating resistances.

(3) The rms error graph for the complex wave digital filter for  $R_s = 1$  and  $R_s = 0$  is worse than that of simple wave digital filters, with approximately the same slope.

(4) The rms error graph for reduced parameter complex wave digital filters with  $R_s = 1$  and  $R_s = 0$  in some cases is slightly worse than that of complex wave digital filter, but with approximately the same slope.

(5) The rms error performance for all wave digital filters with  $R_s = 10$  is approximately the same, after first four or five bits, and the rms error graph tends to follow a straight line. Departure from the straight line is more noticeable for the simple wave digital filter than the complex wave digital filter or the reduced parameter complex wave digital filter for the first six or seven bits.





(6) Although, ideally we expect the two conventional digital filters (i.e. the direct and cascaded digital filters) to have the same rms error, the factorization process used for the denominator introduces some slight error in the pole locations, which in some cases changes the rms error value.

#### D. COMPARISON OF RESULTS WITH EACH OTHER AND WITH THOSE PUBLISHED IN THE LITERATURE

The results obtained in this chapter and in Section C indicate the following points when compared with each other and with results published in literature:

(1) For all purposes the digital filter derived directly from analogue doubly terminated LC structure is the least sensitive structure with respect to filter component values. This fact should not be confused with the sensitivity of the digital filter to polynomial coefficients. The foregoing also apply to band pass or band stop filters derived from analogue low pass LC realization.

(2) It is worthwhile to consider the structure of wave digital filters compared with the structure of analogue LC doubly terminated filters for cases under study (i.e. seventh order low-pass filters). The total number resonances in the analogue LC filter shown in Fig. 5.10, is ten. These are evenly distributed resonances and can be compared with the total number of delay free feedback paths in the wave digital filter, which for the case of simple wave digital filter is eight as shown in Fig. 5.11. Note that the delay free feedbacks are localized towards the source for wave digital filters with no delay free path in port two. Also, as shown in Fig. 5.12, the total number of delay free feedback paths for complex wave digital filters is five, again



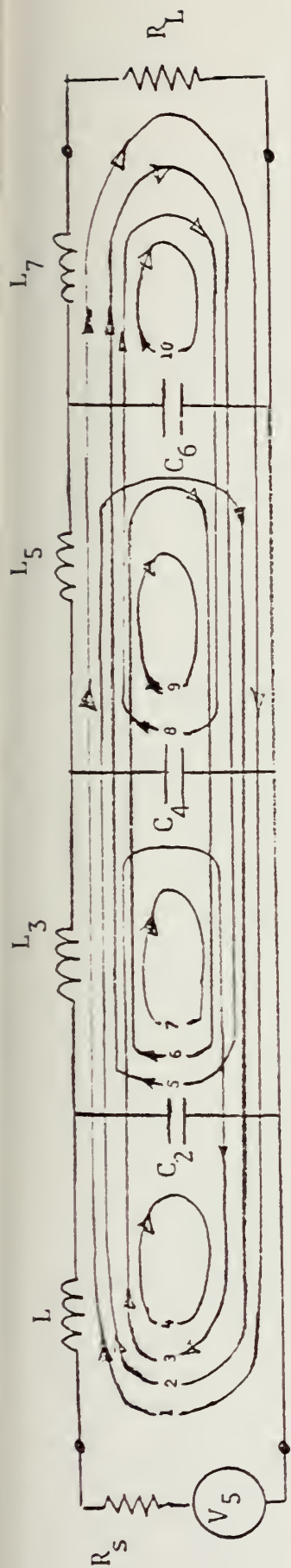


Fig. 5.10. Internal resonances in a 7th order doubly terminated LC filter. Note that the total number of resonances is ten and they are evenly distributed.

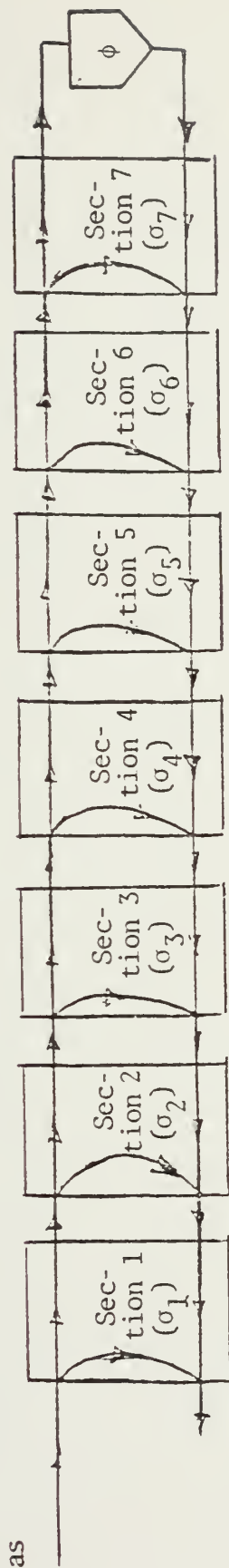


Fig. 5.11. Internal delay free feedback in a wave digital filter patterned after the 7th order doubly terminated LC structure (with no delay free path on port two). Note that the total number of delay free feedback paths is eight and they are localized towards the source end.



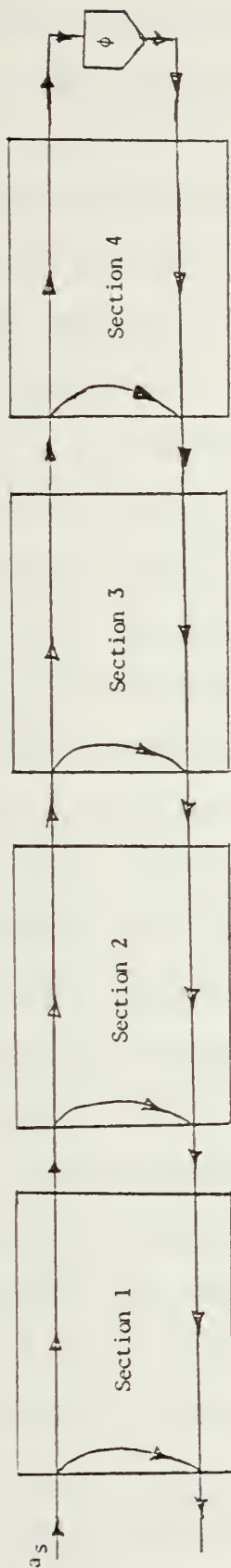


Fig. 5.12. Internal delay free feedback in a complex wave digital filter patterned after the 7th order doubly terminated LC structure (with no delay free path in port two). Note that the total number of delay free feedback paths are five and they are localized towards the source end.



localized towards the source end. Had we designed the wave digital filter with no delay free path in port one this localization would have occurred towards the load end. Now with the above points in mind we can establish the following:

(a) From the results of the large number of different filters studied in this chapter, it is apparent that wave digital filters do exhibit a high sensitivity to terminating resistances. For the case of filters with low terminating source resistances (i.e.  $R_s=0$  and  $R_s=1$ ) the rms error behavior of the wave digital filter is almost identical to direct digital filter derived from the analogue LC structure, because of the localization of the delay free feedback paths towards the low resistance termination. For the case of the filters with high source resistance, the localization of the delay free feedback path occurs towards the high resistance termination; thus the rms error, compared with that of the direct digital filter, is worse; this is due to the localization of the delay free feedback in the high sensitivity port of the filter. Had we designed the high resistive source impedance filters with no delay free path in port one, (thus directing the localization of the delay free feedback paths towards the load end) we would have obtained a better performance.

(b) The rms error performance of simple wave digital filters were considerably better than complex wave digital filters for low source resistances (i.e.  $R_s=0$ ,  $R_s=1$ ). Thus rms error behavior can be explained as follows because there are fewer delay free feedback paths for the complex wave digital filters. This feedback occurs in fewer sections and the error performance gets worse. For high source resistance, i.e.  $R_s=10$ , the rms error performance for all wave digital filters after





fifth or sixth bit becomes almost identical. This can be explained because of the high sensitivity of the wave digital filters to the terminating high source resistance. Thus the localized delay free feedbacks near the source increases the error performance for the simple wave digital filter while for the complex wave digital filter, with fewer feedbacks, the sensitivity to high source impedance drops. Therefore the two algorithms tend to give nearly the same rms error performance.

(c) In some cases especially for lower source impedances the rms error performance of reduced complex wave digital filter is slightly worse than that of complex wave digital filter. This is probably due to the fact that, when the truncation process is done on a large number of multipliers, the effect of truncation tends to be compensating, but if the number of multipliers are reduced, this equalization is less evident.

(3) From the above discussion we can conclude that the ultimate rms error performance of wave digital filters tends to that of the directly digitized analogue resistively terminated LC filters. To achieve this ultimate performance in designing the wave digital filter, we have to use simple sections as much as possible, and also since terminating resistances are the most important we must direct the delay free port of wave digital filters towards the low resistance termination.

(4) If we wanted to make a band pass or band stop filter, which has LC tank circuits, the complex wave digital filter derived from these LC sections would not give the same rms error performance as that of the original conventional digitized LC filter. Moreover the algorithms for complex wave digital filters derived from LC tank circuits in the



literature [2], [3] are of the reduced parameter type complex wave digital filter, which furthermore have poorer error performance as can be seen from the graphs in some cases.

In Chapter VII we discuss briefly how to design a simple wave digital filter algorithm for a given band pass or band stop specification. It is expected that these filters will have better error performance than the complex wave digital filters given in the literature [2] and [3].

(5) it is also interesting to note that, after familiarization with wave digital filter theory, it is much easier to design a wave digital filter derived from analogue LC filters than to design a direct conventional digital filter both derived from the same analogue LC filter. Thus the wave digital filter is a practical design technique. Furthermore in order to reduce the rms error performance of wave digital filters we can even make use of a combination of complex wave digital and simple wave digital algorithms. For example when we have a high source impedance, we can make use of one or two complex sections near the high resistance end, followed by several simple sections later on.



## References

1. Anatol I. Zverev, Handbook of Filter Synthesis, 1967, published by John Wiley and Sons, Inc.
2. S. Erfani, Design of Fixed and Variable Digital Filters Using Generalized Delay Units, Ph.D. Dissertation, 1976, Southern Methodist University, Dallas, Texas.
3. K. S. Thyagarajan, One and Two-dimensional Wave Digital Filters with Low Coefficient Sensitivities, 1977, Doctoral Dissertation, Concordia University, Montreal, Canada.



## VI. DERIVATION OF PARTIAL SENSITIVITY FUNCTIONS OF WAVE DIGITAL FILTERS

### A. INTRODUCTION

The main intent of this chapter is to derive the sensitivity of wave digital filter with the aid of partial differentiation of the filter algorithm with respect to wave digital filter multiplier coefficients, i.e.  $\sigma$ 's and  $\phi$ . Having found these sensitivity functions the sensitivity of wave digital filter with respect to the original filter component values, i.e. L's, C's,  $R_s$ , and  $R_L$  were found. Finally the behavior of these two sensitivity functions are investigated and compared with each other in the frequency domain and with the results obtained in Chapter V.

### B. DERIVATION OF THE SENSITIVITY FUNCTION OF WAVE DIGITAL FILTER DUE TO VARIATION IN MULTIPLIER COEFFICIENTS

The general wave digital algorithm for the third order filter given in Fig. 6.2, with no delay free path in port two, is of the form of equation (6.1)

$$\begin{bmatrix} a_{11}(z) \\ b_{11}(z) \end{bmatrix} = \begin{bmatrix} f_1(\sigma_1, z) \\ f_2(\sigma_2, z) \\ f_3(\sigma_3, z) \end{bmatrix} \begin{bmatrix} b_{32}(z) \\ \phi b_{32}(z) \end{bmatrix} \quad (6.1)$$

where  $f_1$ ,  $f_2$ , and  $f_3$  are 2x2 matrices of the general form of

$$f_1(\sigma_1, z) = \begin{bmatrix} \frac{\alpha_1 + \beta_1 z^{-1}}{1 + \gamma_1 z^{-1}} & \frac{\alpha_2 + \beta_2 z^{-1}}{1 + \gamma_2 z^{-1}} \\ \frac{\alpha_3 + \beta_3 z^{-1}}{1 + \gamma_3 z^{-1}} & \frac{\alpha_4 + \beta_4 z^{-1}}{1 + \gamma_4 z^{-1}} \end{bmatrix}$$





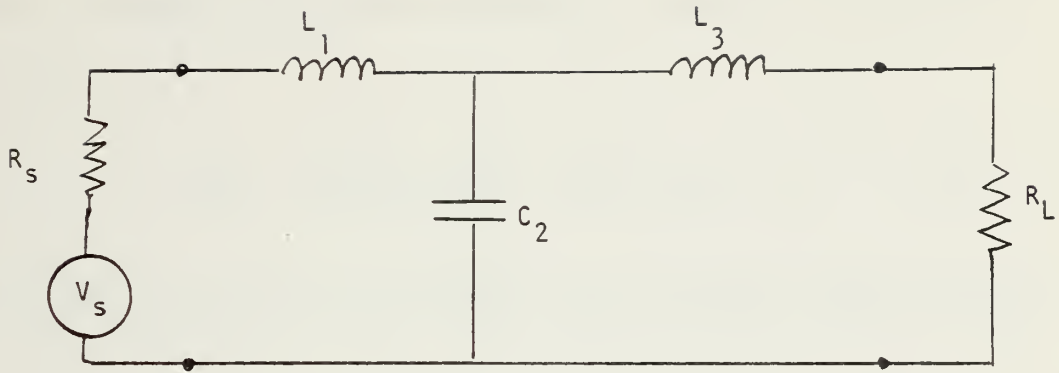


Fig. 6.1. A general third order doubly terminated analogue low pass LC filter.

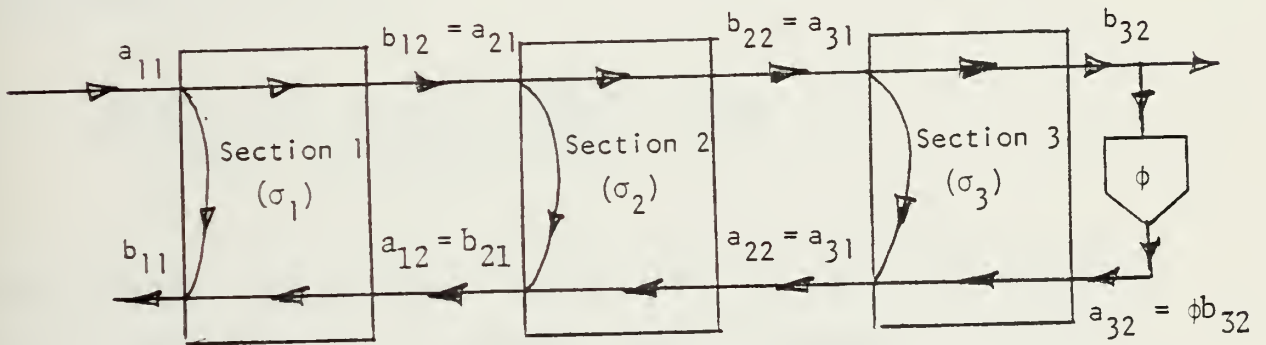


Fig. 6.2. The wave digital filter derived from the analogue LC filter of Fig. 6.1 with no delay free path on port two.



where  $\alpha_i, \beta_i, \gamma_i$  are functions of  $\sigma_1$ .

From Fig. 6.2 the transfer function of the filter in the time domain as depicted in Chapter IV and also explained in Appendix 1A with unity impulse input will be

$$h(nT) = \left(\frac{1+\phi}{2}\right) b_{32}(nT) = h(\sigma_1, \sigma_2, \dots, \phi) \quad (6.2)$$

Thus the transfer function of the filter in the frequency domain will be

$$H(\sigma_1, \sigma_2, \dots, \phi) = \frac{1+\phi}{2} \sum_{n=0}^N b_{32}(nT) e^{-j\omega nT}$$

and the sensitivity of the filter due to variations in  $\sigma_1$  in the frequency domain is given by

$$\frac{\partial H(\sigma_1, \sigma_2, \dots, \phi)}{\partial \sigma_1} = \frac{1+\phi}{2} \sum_{n=0}^N \frac{\partial b_{32}(nT)}{\partial \sigma_1} e^{-j\omega nT}$$

Note that from equation (6.1), the matrices  $f_2, f_3$  are not functions of  $\sigma_1$ . Thus the derivative of equation (6.1) with respect to  $\sigma_1$  will be

$$\begin{bmatrix} 0 \\ \frac{\partial b_{11}(z)}{\partial \sigma_1} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1(\sigma_1, z)}{\partial \sigma_1} \\ f_2(\sigma_2, z) \\ f_3(\sigma_3, z) \end{bmatrix} \begin{bmatrix} \frac{\partial b_{32}(z)}{\partial \sigma_1} \\ \phi \cdot \frac{\partial b_{32}(z)}{\partial \sigma_1} \end{bmatrix} \quad (6.3)$$

Note  $a_{11}(z)$  is input and is independent of  $\sigma_1$  also.



A closer look at equation (6.2) reveals that with unity impulse input, the time domain sensitivity function of the filter with respect to  $\sigma_1$  is

$$\frac{\partial h(\sigma_1, \sigma_2, \dots, \phi)}{\partial \sigma_1} = \left(\frac{1+\phi}{2}\right) \frac{\partial b_{32}}{\partial \sigma_1}$$

Thus to find the sensitivity function of the filter with respect to  $\sigma_1$  all we have to do is write the complete iterative equations of the filter from equation (6.1) and then differentiate them with respect to  $\sigma_1$ . Note that in doing so actually all the iterative equations derived from matrices  $f_2(\sigma_2, z)$ ,  $f_3(\sigma_3, z)$  will not be effected and the only iterative equations being differentiated will be that of matrix  $f_1(\sigma_1, z)$ . So in general we can find  $\frac{\partial H}{\partial \sigma_n}$ , i.e. the sensitivity of the wave digital filter with respect to all wave digital filter multiplier coefficients  $\sigma_n$ . For the special case of  $\frac{\partial H}{\partial \phi}$  from equation (6.2) we have

$$\frac{\partial h(nT)}{\partial \phi} = \frac{1}{2} (b_{32}(nT) + \phi \frac{\partial b_{32}}{\partial \phi})$$

and the filter sensitivity function with respect to  $\phi$  in the frequency domain will be

$$\frac{\partial H(\sigma_1, \sigma_2, \dots, \phi)}{\partial \phi} = \sum_{n=0}^N \left[ \frac{b_{32}(nT)}{2} + \frac{(1+\phi)}{2} \cdot \frac{\partial b_{32}}{\partial \phi} \right] e^{-j\omega nT}$$

The above procedure can best be illustrated with the aid of a simple example.

#### Example 1

Given the third order wave digital filter of Fig. 6.2 which is derived from the doubly terminated LC filter of Fig. 6.1; it is required



to find the sensitivity function of the filter with respect to  $\sigma_1$  in the frequency domain. The wave digital filter is with no delay free path in port two.

### Procedure

The iterative equations of the wave digital filter of Fig. 6.2 in the time domain with the aid of Table 4.2b and with all initial conditions set to zero are:

$$\begin{aligned}
 b_{12}(n) &= a_{11}(n) + a_{11}(n-1) - a_{12}(n-1) + \sigma_1(a_{12}(n-1) - b_{12}(n-1)) \\
 a_{21}(n) &= b_{12}(n) \\
 b_{22}(n) &= a_{22}(n-1) + \sigma_2(a_{21}(n) + a_{21}(n-1) - a_{22}(n-1) - b_{22}(n-1)) \\
 a_{31}(n) &= b_{22}(n) \\
 b_{32}(n) &= a_{31}(n) + a_{31}(n-1) - a_{32}(n-1) + \sigma_3(a_{32}(n-1) - b_{32}(n-1)) \\
 a_{32}(n) &= b_{32}(n) \cdot \phi \\
 b_{31}(n) &= a_{31}(n) + \sigma_3(a_{32}(n) - a_{31}(n) + a_{32}(n-1) - b_{31}(n-1)) \\
 a_{22}(n) &= b_{31}(n) \\
 b_{21}(n) &= a_{22}(n) - a_{21}(n) + a_{22}(n-1) + \sigma_2(a_{21}(n) - b_{21}(n-1)) \\
 a_{12}(n) &= b_{21}(n) \\
 b_{11}(n) &= a_{11}(n) + \sigma_1(a_{12}(n) - a_{11}(n) + a_{12}(n-1) - b_{11}(n-1))
 \end{aligned} \tag{6.4}$$

Note that to find the transfer function of the filter in the time domain, the input  $a_{11}(n)$  will be

$$a_{11}(n) = \begin{cases} 1.0 & n=0 \\ 0 & n \neq 0 \end{cases}$$





and the iterative equations after partial differentiation with respect to  $\sigma_1$  are

$$\frac{\partial b_{12}(n)}{\partial \sigma_1} = - \frac{\partial a_{12}(n-1)}{\partial \sigma_1} + a_{12}(n-1) - b_{12}(n-1) + \sigma_1 \left( \frac{\partial a_{12}(n-1)}{\partial \sigma_1} - \frac{\partial b_{12}(n-1)}{\partial \sigma_1} \right)$$

$$\frac{\partial a_{21}(n)}{\partial \sigma_1} = \frac{\partial b_{12}(n)}{\partial \sigma_1} \quad (6.5)$$

$$\frac{\partial b_{22}(n)}{\partial \sigma_1} = \frac{\partial a_{22}(n-1)}{\partial \sigma_1} + \sigma_2 \left( \frac{\partial a_{21}(n)}{\partial \sigma_1} + \frac{\partial a_{21}(n-1)}{\partial \sigma_1} - \frac{\partial a_{22}(n-1)}{\partial \sigma_1} - \frac{\partial b_{22}(n-1)}{\partial \sigma_1} \right)$$

etc.  
 $\vdots$   
 $\vdots$

and in this way, we differentiate all the equations in equation (6.4) with respect to  $\sigma_1$  up to

$$\frac{\partial a_{12}(n)}{\partial \sigma_1} = \frac{\partial b_{21}(n)}{\partial \sigma_1}$$

$$\frac{\partial b_{11}}{\partial \sigma_1} = a_{12}(n) - a_{11}(n) + a_{12}(n-1) - b_{11}(n-1) + \sigma_1 \left( \frac{\partial a_{12}(n)}{\partial \sigma_1} + \frac{\partial a_{12}(n-1)}{\partial \sigma_1} - \frac{\partial b_{11}(n-1)}{\partial \sigma_1} \right)$$

Thus the sensitivity function of the filter in the time domain from equation 6.2 is

$$\frac{\partial h(nT)}{\partial \sigma_1} = \left( \frac{1+\phi}{2} \right) \cdot \frac{\partial b_{32}(nT)}{\partial \sigma_1}$$

and the sensitivity function of the filter in the frequency domain will be

$$\frac{\partial H(\omega)}{\partial \sigma_1} = \sum_{n=0}^N \frac{\partial h(nT)}{\partial \sigma_1} e^{-j\omega nT}$$



where  $T$  is the sampling interval,  $\omega$  is the frequency of input signal, and  $N$  is chosen large enough for transients due to the impulse response to decay.

C. DERIVATION OF SENSITIVITY FUNCTION OF WAVE DIGITAL FILTER IN THE FREQUENCY DOMAIN WITH RESPECT TO ITS ORIGINAL COMPONENT VALUES, I.E.  $L$ 's,  $C$ 's,  $R_s$ , AND  $R_L$

As noted in Table 4.1 and 4.2 and mentioned in Appendix 1-A, there exists a one to one relationship between the wave digital multiplier coefficients, i.e.  $\sigma$ 's and the reflection coefficient  $\phi$  of the wave digital filter, with the original filter component values, i.e.  $L$ 's,  $C$ 's,  $R_s$ , and  $R_L$ . Thus in order to find the sensitivity of the wave digital filter algorithm with respect to original filter component values, the best way is to find the sensitivity of the wave digital filter with respect to wave digital filter multiplier coefficients, i.e.  $\frac{\partial H(\sigma_1, \sigma_2, \dots, \phi)}{\partial \sigma_1}$ , etc. and from these, the sensitivity of the wave digital filter with respect to original filter component values, i.e.  $\frac{\partial H(L_1, C_2, L_3, \dots, R_s, R_L)}{\partial L_1}$ , etc. can be found. This procedure can best be illustrated with the aid of an example, the results of which will be used later on to find the semi-logarithmic sensitivity (or percentage sensitivity; or normalized sensitivity) of a seventh order low pass resistively terminated wave digital LC filter with respect to original components (i.e.  $L$ 's,  $C$ 's,  $R_s$ , and  $R_L$ ). Later on these normalized sensitivity functions will be compared with that of the normalized sensitivity functions of the wave digital filter with respect to wave digital filter multiplier coefficients (i.e.  $\sigma$ 's and  $\phi$ ).



### Example 2

Given the sensitivity functions of the seventh order low pass wave digital filter of Fig. 6.4, designed with no delay free path in port two from the original LC structure of Fig. 6.3; it is required to find the sensitivity function of the wave digital filter with respect to original wave digital filter component values.

### Procedure

Since the wave digital filter is of the type with no delay free path in port two, from Table 4.2 we have

$$\sigma_1 = \frac{R_1}{R_2} = \frac{R_1}{R_1 + L_1} = f(R_s, L_1) \quad (6.6)$$

where  $R_1 = R_s$ .

Also at this stage it is important to note that L and C components of the filter used in equation (6.6) and in the remainder of the example are predivided or premultiplied by the factor  $\tan \frac{\Omega_{cd} T}{2} / \omega_c$  in order to take into account the effect of the sampling period and also the critical frequency of the digital filter, where  $\Omega_{cd}$  is the critical frequency of the digital filter in rad/sec and  $\omega_c$  is the critical frequency of the analogue LC filter in rad/sec and T is the sampling time in sec. This factor is obtained by the multiplication of the prewarping factor, i.e.  $2 \tan \frac{\Omega_{cd} T}{2} / T \omega_c$  by the sampling factor of the reactive components of the wave digital filter, i.e. T/2. The factor  $\tan \frac{\Omega_{cd} T}{2} / \omega_c$  reduces to  $\tan \frac{\Omega_{cd} T}{2}$  when the critical frequency of the original analogue filter is normalized at  $\omega_c = 1$  rad/sec.



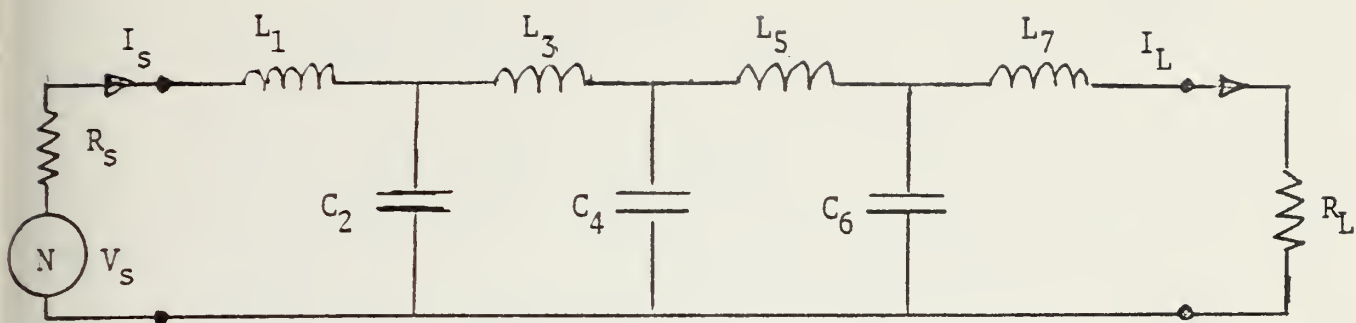


Fig. 6.3. Seventh order low pass analogue double resistively terminated LC filter.

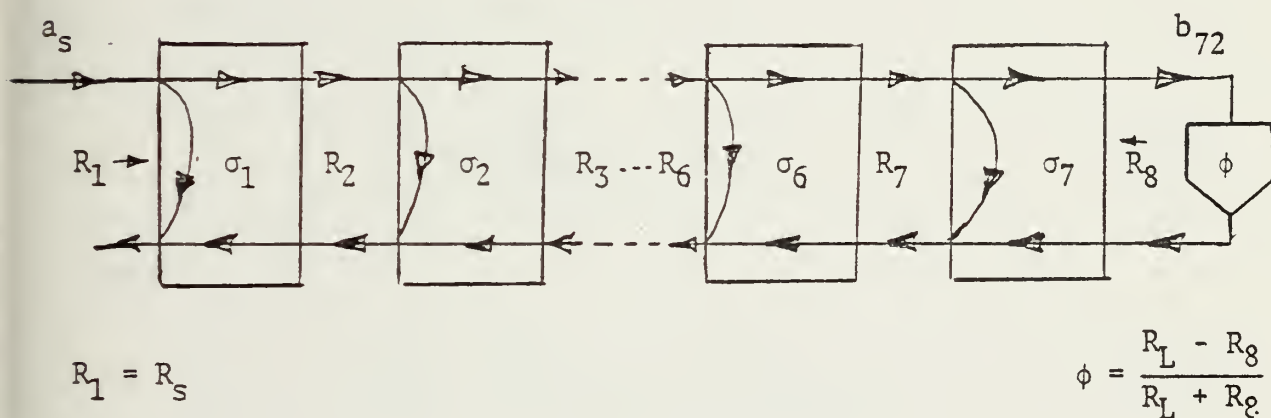


Fig. 6.4. The wave digital filter derived from the seventh order analogue LC filter of Fig. 6.3 with no delay free path on port two.





Thus from equation (6.6)

$$G_2 = \sigma_1 G_1 \quad (6.7)$$

where

$$G_i = \frac{1}{R_i}, \quad i = 1, 2, 3, \dots$$

Also

$$b_2 = \frac{G_2}{G_3} = \frac{G_2}{G_2 + C_2} \quad (6.8)$$

and

$$R_3 = \sigma_2 R_2$$

Also from equation (6.7) and (6.8) we have

$$\sigma_2 = \frac{\sigma_1 G_1}{\sigma_1 G_1 + C_2} = f(\sigma_1, C_2) \quad (6.9)$$

thus

$$\frac{\partial \sigma_2}{\partial \sigma_1} = \frac{G_1 C_2}{(\sigma_1 G_1 + C_2)^2} = \frac{C_2 R_1}{(\sigma_1 + R_1 C_2)^2} \quad (6.10)$$

In the same way

$$\sigma_3 = \frac{R_3}{R_3 + L_3} = \frac{\sigma_2 R_2}{\sigma_2 R_2 + L_3} = f(\sigma_2, L_3) \quad (6.11)$$

$$\frac{\partial \sigma_3}{\partial \sigma_2} = \frac{R_2 L_3}{(\sigma_2 R_2 + L_3)^2} \quad (6.12)$$



$$\sigma_4 = \frac{G_4}{G_4 + C_4} = \frac{\sigma_3 G_3}{\sigma_3 G_3 + C_4} = f(\sigma_3, C_4) \quad (6.13)$$

$$\frac{\partial \sigma_4}{\partial \sigma_3} = \frac{G_3 C_4}{(\sigma_3 G_3 + C_4)^2} \quad (6.14)$$

$$\sigma_5 = \frac{R_5}{R_4 + L_5} = \frac{\sigma_4 R_4}{\sigma_4 R_4 + L_5} = f(\sigma_4, L_5) \quad (6.15)$$

$$\frac{\partial \sigma_5}{\partial \sigma_4} = \frac{R_4 L_5}{(\sigma_4 R_4 + L_5)^2} \quad (6.16)$$

$$\sigma_6 = \frac{G_6}{G_6 + C_6} = \frac{\sigma_5 G_5}{\sigma_5 G_5 + C_6} = f(\sigma_5, C_6) \quad (6.17)$$

$$\frac{\partial \sigma_6}{\partial \sigma_5} = \frac{G_5 C_6}{(\sigma_5 G_5 + C_6)^2} \quad (6.18)$$

$$\sigma_7 = \frac{R_7}{R_7 + L_7} = \frac{\sigma_6 R_6}{\sigma_6 R_6 + L_7} = f(\sigma_6, L_7) \quad (6.19)$$

$$\frac{\partial \sigma_7}{\partial \sigma_6} = \frac{R_6 L_7}{(\sigma_6 R_6 + L_7)^2} \quad (6.20)$$

$$\phi = \frac{R_L - R_8}{R_L + R_8} = \frac{R_L - G_7 \sigma_7}{R_L + G_7 \sigma_7} = f(\sigma_7, R_L) \quad (6.21)$$

$$\frac{\partial \phi}{\partial \sigma_7} = \frac{-2G_7 R_L}{(R_L + G_7 \sigma_7)^2} \quad (6.22)$$

Note that from equation (6.21) it can be seen that

$$\phi = f(L_1, C_2, L_3, C_4, \dots, L_7, R_S, R_L)$$



Note also that the transfer function  $H$  of the filter is a function of  $\sigma_1, \sigma_2$ , etc., i.e.

$$H = f(\sigma_1, \sigma_2, \dots, \sigma_7, \phi) \quad (6.23)$$

and also

$$H = f(L_1, C_2, L_3, \dots, L_7, R_s, R_L) \quad (6.24)$$

Thus if we have  $\frac{\partial H}{\partial \sigma_1}$ , we can derive  $\frac{\partial H}{\partial R_s}$  from (6.23) by writing all partials of  $\frac{\partial H}{\partial R_s}$  from equation (6.23), i.e.

$$\begin{aligned} \frac{\partial H}{\partial R_s} &= \frac{\partial H}{\partial \sigma_1} \cdot \frac{\partial \sigma_1}{\partial R_s} + \frac{\partial H}{\partial \sigma_2} \cdot \frac{\partial \sigma_2}{\partial R_s} + \frac{\partial H}{\partial \sigma_3} \cdot \frac{\partial \sigma_3}{\partial R_s} + \\ &\quad \frac{\partial H}{\partial \sigma_4} \cdot \frac{\partial \sigma_4}{\partial R_s} + \frac{\partial H}{\partial \sigma_5} \cdot \frac{\partial \sigma_5}{\partial R_s} + \frac{\partial H}{\partial \sigma_6} \cdot \frac{\partial \sigma_6}{\partial R_s} + \\ &\quad \frac{\partial H}{\partial \sigma_7} \cdot \frac{\partial \sigma_7}{\partial R_s} + \frac{\partial H}{\partial \phi} \cdot \frac{\partial \phi}{\partial R_s} \end{aligned} \quad (6.25)$$

$$\begin{aligned} &= A_1 \cdot \frac{\partial H}{\partial \sigma_1} + A_2 \cdot \frac{\partial H}{\partial \sigma_2} + A_3 \cdot \frac{\partial H}{\partial \sigma_3} + A_4 \cdot \frac{\partial H}{\partial \sigma_4} \\ &+ A_5 \cdot \frac{\partial H}{\partial \sigma_5} + A_6 \cdot \frac{\partial H}{\partial \sigma_6} + A_7 \cdot \frac{\partial H}{\partial \sigma_7} + A_8 \cdot \frac{\partial H}{\partial \sigma_8} \end{aligned} \quad (6.26)$$

to find  $A_1, A_2, A_3$ , etc. we have from equation (6.6) with  $R_1 = R_s$ .

$$A_1 = \frac{\partial \sigma_1}{\partial R_s} = \frac{L_1}{(R_1 + L_1)^2} \quad (6.27)$$

and from equation (6.25) and the chain rule

$$A_2 = \frac{\partial \sigma_2}{\partial R_s} = \frac{\partial \sigma_2}{\partial \sigma_1} \cdot \frac{\partial \sigma_1}{\partial R_s}$$



Thus from equations (6.10) and (6.27) we have

$$A_2 = A_1 \cdot \frac{C_2 R_1}{(\sigma_1 + C_2 R_1)^2} \quad (6.28)$$

and in the same way

$$A_3 = \frac{\partial \sigma_3}{\partial R_s} = A_2 \cdot \frac{R_2 L_3}{(\sigma_2 R_2 + L_3)^2} \quad (6.29)$$

$$A_4 = \frac{\partial \sigma_4}{\partial R_s} = A_3 \cdot \frac{G_3 C_4}{(\sigma_3 G_3 + C_4)^2} \quad (6.30)$$

$$A_5 = \frac{\partial \sigma_5}{\partial R_s} = A_4 \cdot \frac{R_4 L_5}{(\sigma_4 R_4 + L_5)^2} \quad (6.31)$$

$$A_6 = \frac{\partial \sigma_6}{\partial R_s} = A_5 \cdot \frac{G_5 C_6}{(\sigma_5 G_5 + C_6)^2} \quad (6.32)$$

$$A_7 = \frac{\partial \sigma_7}{\partial R_s} = A_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6 + L_7)^2} \quad (6.33)$$

$$A_8 = \frac{\partial \phi}{\partial R_s} = A_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7 + R_L)^2} \quad (6.34)$$

Thus from equations (6.26) to (6.34) we can find the sensitivity of the wave digital filter with respect to  $R_s = R_1$ . To find the sensitivity of wave digital filter with respect to  $L_1$ , i.e.  $\frac{\partial H}{\partial L_1}$  from equation (6.24) we can write





$$\begin{aligned}\frac{\partial H}{\partial L_1} &= \frac{\partial H}{\partial \sigma_1} \cdot \frac{\partial \sigma_1}{\partial L_1} + \frac{\partial H}{\partial \sigma_2} \cdot \frac{\partial \sigma_2}{\partial L_1} + \frac{\partial H}{\partial \sigma_3} \cdot \frac{\partial \sigma_3}{\partial L_1} + \frac{\partial H}{\partial \sigma_4} \cdot \frac{\partial \sigma_4}{\partial L_1} \\ &+ \frac{\partial H}{\partial \sigma_5} \cdot \frac{\partial \sigma_5}{\partial L_1} + \frac{\partial H}{\partial \sigma_6} \cdot \frac{\partial \sigma_6}{\partial L_1} + \frac{\partial H}{\partial \sigma_7} \cdot \frac{\partial \sigma_7}{\partial L_1} + \frac{\partial H}{\partial \phi} \cdot \frac{\partial \phi}{\partial L_1}\end{aligned}\quad (6.35)$$

$$\begin{aligned}&= B_1 \cdot \frac{\partial H}{\partial \sigma_1} + B_2 \cdot \frac{\partial H}{\partial \sigma_2} + B_3 \cdot \frac{\partial H}{\partial \sigma_3} + B_4 \cdot \frac{\partial H}{\partial \sigma_4} \\ &\quad + B_5 \cdot \frac{\partial H}{\partial \sigma_5} + B_6 \cdot \frac{\partial H}{\partial \sigma_6} + B_7 \cdot \frac{\partial H}{\partial \sigma_7} + B_8 \cdot \frac{\partial H}{\partial \phi}\end{aligned}\quad (6.36)$$

To find  $B_1, B_2, B_3$ , etc. we have from equations (6.6), and (6.36)

$$B_1 = \frac{\partial \sigma_1}{\partial L_1} = \frac{-R_1}{(R_1 + L_1)^2} \quad (6.37)$$

and from equation (6.35) and the Chain Rule

$$B_2 = \frac{\partial \sigma_2}{\partial L_1} = \frac{\partial \sigma_2}{\partial \sigma_1} \cdot \frac{\partial \sigma_1}{\partial L_1}$$

Thus from equations (6.10) and (6.37)

$$B_2 = B_1 \cdot \frac{G_1 C_2}{(\sigma_1 G_1 + C_2)^2} = \frac{-C_2 R_1^2}{(R_1 + L_1)^2 (\sigma_1 + R_1 C_2)^2} \quad (6.38)$$

and in the same way

$$B_3 = \frac{\partial \sigma_3}{\partial L_1} = B_2 \cdot \frac{R_2 L_3}{(\sigma_2 R_2 + L_3)^2} \quad (6.39)$$

$$B_4 = \frac{\partial \sigma_4}{\partial L_1} = B_3 \cdot \frac{G_3 C_4}{(\sigma_3 G_3 + C_4)^2} \quad (6.40)$$



$$B_5 = \frac{\partial \sigma_5}{\partial L_1} = B_4 \cdot \frac{R_4 L_5}{(\sigma_4 R_4 + L_5)^2} \quad (6.41)$$

$$B_6 = \frac{\partial \sigma_6}{\partial L_1} = B_5 \cdot \frac{G_5 C_6}{(\sigma_5 G_5 + C_6)^2} \quad (6.42)$$

$$B_7 = \frac{\partial \sigma_7}{\partial L_1} = B_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6 + L_7)^2} \quad (6.43)$$

$$B_8 = \frac{\partial \phi}{\partial L_1} = B_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7 + R_L)^2} \quad (6.44)$$

Thus from equations (6.36) to (6.44) we can find the sensitivity of the wave digital filter with respect to  $L_1$ .

To find the sensitivity of wave digital filter with respect to  $C_2$ , i.e.  $\frac{\partial H}{\partial C_2}$  from equation (6.24) we can write

$$\begin{aligned} \frac{\partial H}{\partial C_2} = & \frac{\partial H}{\partial \sigma_1} \cdot \frac{\partial \sigma_1}{\partial C_2} + \frac{\partial H}{\partial \sigma_2} \cdot \frac{\partial \sigma_2}{\partial C_2} + \frac{\partial H}{\partial \sigma_3} \cdot \frac{\partial \sigma_3}{\partial C_2} + \frac{\partial H}{\partial \sigma_4} \cdot \frac{\partial \sigma_4}{\partial C_2} \\ & + \frac{\partial H}{\partial \sigma_5} \cdot \frac{\partial \sigma_5}{\partial C_2} + \frac{\partial H}{\partial \sigma_6} \cdot \frac{\partial \sigma_6}{\partial C_2} + \frac{\partial H}{\partial \sigma_7} \cdot \frac{\partial \sigma_7}{\partial C_2} + \frac{\partial H}{\partial \sigma} \cdot \frac{\partial \phi}{\partial C_2} \end{aligned} \quad (6.45)$$

or

$$\begin{aligned} \frac{\partial H}{\partial C_2} = & D_1 \cdot \frac{\partial H}{\partial \sigma_1} + D_2 \cdot \frac{\partial H}{\partial \sigma_2} + D_3 \cdot \frac{\partial H}{\partial \sigma_3} + D_4 \cdot \frac{\partial H}{\partial \sigma_4} + D_5 \cdot \frac{\partial H}{\partial \sigma_5} \\ & + D_6 \cdot \frac{\partial H}{\partial \sigma_6} + D_7 \cdot \frac{\partial H}{\partial \sigma_7} + D_8 \cdot \frac{\partial H}{\partial \sigma_8} \end{aligned} \quad (6.46)$$

Note that the coefficient of the first term, i.e.  $D_1 = 0$ .

From equation (6.9)



$$D_2 = \frac{\partial \sigma_2}{\partial C_2} = \frac{-G_2}{(G_2 + C_2)^2} \quad (6.47)$$

and from equation (6.45) and the Chain Rule

$$D_3 = \frac{\partial \sigma_3}{\partial C_2} = \frac{\partial \sigma_3}{\partial \sigma_2} \cdot \frac{\partial \sigma_2}{\partial C_2} \quad (6.48)$$

Thus from equations (6.12) and (6.47)

$$D_3 = D_2 \cdot \frac{R_2 L_3}{(\sigma_2 R_2 + L_3)^2} \quad (6.49)$$

and in the same way

$$D_4 = D_3 \cdot \frac{G_3 C_4}{(\sigma_3 G_3 + C_4)^2} \quad (6.50)$$

$$D_5 = D_4 \cdot \frac{R_4 L_5}{(\sigma_4 R_4 + L_5)^2} \quad (6.51)$$

$$D_6 = D_5 \cdot \frac{G_5 C_6}{(\sigma_5 G_5 + C_6)^2} \quad (6.52)$$

$$D_7 = D_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6 + L_7)^2} \quad (6.53)$$

$$D_8 = D_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7 + R_L)^2} \quad (6.54)$$

Thus from equations (6.46) to (6.54) we can find the sensitivity of the wave digital filter with respect to  $C_2$ .

To find the sensitivity of wave digital filter with respect to  $L_3, C_4, L_5, C_6, L_7$  in the same way we have



$$\frac{\partial H}{\partial L_3} = E_3 \cdot \frac{\partial H}{\partial \sigma_3} + E_4 \cdot \frac{\partial H}{\partial \sigma_4} + E_5 \cdot \frac{\partial H}{\partial \sigma_5} + E_6 \cdot \frac{\partial H}{\partial \sigma_6} + E_7 \cdot \frac{\partial H}{\partial \sigma_7} + E_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.55)$$

and

$$\frac{\partial H}{\partial C_4} = P_4 \cdot \frac{\partial H}{\partial \sigma_4} + P_5 \cdot \frac{\partial H}{\partial \sigma_5} + P_6 \cdot \frac{\partial H}{\partial \sigma_6} + P_7 \cdot \frac{\partial H}{\partial \sigma_7} + P_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.56)$$

$$\frac{\partial H}{\partial L_5} = Q_5 \cdot \frac{\partial H}{\partial \sigma_5} + Q_6 \cdot \frac{\partial H}{\partial \sigma_6} + Q_7 \cdot \frac{\partial H}{\partial \sigma_7} + Q_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.57)$$

$$\frac{\partial H}{\partial C_6} = S_6 \cdot \frac{\partial H}{\partial \sigma_6} + S_7 \cdot \frac{\partial H}{\partial \sigma_7} + S_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.58)$$

$$\frac{\partial H}{\partial L_7} = U_7 \cdot \frac{\partial H}{\partial \sigma_7} + U_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.59)$$

$$\frac{\partial H}{\partial R_L} = V_8 \cdot \frac{\partial H}{\partial \phi} \quad (6.60)$$

where

$$E_3 = \frac{-R_3}{(R_3 + L_3)^2} \quad (6.61)$$

$$E_4 = E_3 \cdot \frac{G_3 C_4}{(\sigma_3 G_3 + C_4)^2} \quad (6.62)$$

$$E_5 = E_4 \cdot \frac{R_4 L_5}{(\sigma_4 R_4 + L_5)^2} \quad (6.63)$$

$$E_6 = E_5 \cdot \frac{G_5 C_6}{(\sigma_5 G_5 + C_6)^2} \quad (6.64)$$

$$E_7 = E_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6 + L_7)^2} \quad (6.65)$$

$$E_8 = E_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7 + R_L)^2} \quad (6.66)$$





and

$$P_4 = \frac{-G_4}{(G_5 + C_4)^2} \quad (6.67)$$

$$P_5 = P_4 \cdot \frac{R_4 L_5}{(\sigma_4 R_4 + L_5)^2} \quad (6.68)$$

$$P_6 = P_5 \cdot \frac{G_5 C_6}{(\sigma_5 G_5 + C_6)^2} \quad (6.69)$$

$$P_7 = P_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6 + L_7)^2} \quad (6.70)$$

$$P_8 = P_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7 + R_L)^2} \quad (6.71)$$

and

$$Q_5 = \frac{-R_5}{(R_5 + L_5)^2} \quad (6.72)$$

$$Q_6 = Q_5 \cdot \frac{G_5 C_6}{(\sigma_5 G_5 + C_6)^2} \quad (6.73)$$

$$Q_7 = Q_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6 + L_7)^2} \quad (6.74)$$

$$Q_8 = Q_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7 + R_L)^2} \quad (6.75)$$

and

$$S_6 = \frac{-G_6}{(G_6 + C_6)^2} \quad (6.76)$$

$$S_7 = S_6 \cdot \frac{R_6 L_7}{(\sigma_6 R_6 + L_7)^2} \quad (6.77)$$

$$S_8 = S_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7 + R_L)^2} \quad (6.78)$$

and

$$U_7 = \frac{-R_7}{(R_7 + L_7)^2} \quad (6.79)$$

$$U_8 = U_7 \cdot \frac{-2G_7 R_L}{(\sigma_7 G_7 + R_L)^2} \quad (6.80)$$



and

$$V_8 = \frac{2G_7\sigma_7}{(\sigma_7 G_7 + R_L)^2} \quad (6.81)$$

Thus we can find the sensitivity of wave digital filter with respect to the original filter component values.

#### D. EXPERIMENTAL STUDY ON THE INTERNAL SENSITIVITY BEHAVIOR OF WAVE DIGITAL FILTERS

In order to analyze the internal sensitivity behavior of wave digital filters with respect to the multiplier coefficients and compare them to the sensitivity of wave digital filters with respect to the original filter component values, in total nine different seventh order low pass filters were taken from the Handbook of Filter Synthesis [1] with source resistances varying from 0 to 10 ohms. The sensitivity behavior of these filters in the frequency domain with respect to both wave digital filter multiplier coefficients and the original filter component values were found using the procedures set in examples 1 and 2. These filters were

- i) seventh order .5 db ripple low pass Chebyshev filter with  $R_s=1.0$
- ii) seventh order .1 db ripple low pass Chebyshev filter with  $R_s=1.0$
- iii) seventh order Butterworth low pass filter with  $R_s=1.0$
- iv) seventh order .5 db ripple low pass Chebyshev filter with  $R_s=10.0$
- v) seventh order .1 db ripple low pass Chebyshev filter with  $R_s=10.0$
- vi) seventh order Butterworth low pass filter with  $R_s=10.0$
- vii) seventh order .5 db ripple low pass Chebyshev filter with  $R_s=0.0$
- viii) seventh order .1 db ripple low pass Chebyshev filter with  $R_s=0.0$
- ix) seventh order Butterworth low pass filter with  $R_s = 0.0$

Note that these filters are the same filters investigated in the sensitivity analysis of Chapter V. In order to make the comparison between the two sensitivity functions, i.e. sensitivity with respect to wave



digital filter multiplier coefficients and sensitivity with respect to wave digital filter original components more meaningful, the semi-logarithmic sensitivity function which is in fact normalized or percentage change in sensitivity is used. These normalized sensitivity functions are  $\frac{\partial H}{\partial \sigma_n} \cdot \sigma_n$  and are plotted on the same graph in the frequency domain in Figs. 6.3 to 6.11.

Note that there are nine variables for the seventh order filter in terms of original components (i.e.  $L_1, C_2, L_3, C_4, L_5, C_6, L_7, R_s$ , and  $R_L$ ) while there are eight variables for the filter with no delay free path in port two (i.e.  $\sigma_1, \sigma_2, \dots, \sigma_7$  and  $\phi$ ), since the source reflection coefficient,  $\theta$  is made equal to zero. Thus the curves of  $\frac{\partial H}{\partial R_s} \cdot R_s$  are plotted on a single graph. The remainder of eight variables are plotted on the same coordinates, i.e.  $\frac{\partial H}{\partial L_1} \cdot L_1$  is plotted on the same coordinate as  $\frac{\partial H}{\partial \sigma_1} \cdot \sigma_1$  and so on. Finally  $\frac{\partial H}{\partial R_L} \cdot R_L$  is plotted with  $\frac{\partial H}{\partial \phi} \cdot \phi$  on the same coordinates. It is important to note that for the filters with  $R_s = 0$ , the normalized partial sensitivity  $\frac{\partial H}{\partial \sigma_1} \cdot \sigma_1$  equals to zero since  $\sigma_1 = 0$  when  $R_L = R_s = 0$ . Also we note that for this special case  $\frac{\partial H}{\partial L_1} = 0$  since in equation (6.26) all the coefficients of  $\frac{\partial H}{\partial L_1}$  are zero with  $R_L = R_s = 0$ . Thus for these reasons for when  $R_s = 0$  we have plotted only  $\frac{\partial H}{\partial \sigma_1}$  rather than  $\frac{\partial H}{\partial \sigma_1} \cdot \sigma_1$  and we also note that  $\frac{\partial H}{\partial L_1}$  is zero for all frequencies.

#### E. ANALYSIS OF THE INTERNAL STRUCTURE OF THE WAVE DIGITAL FILTERS IN FREQUENCY DOMAIN USING FILTER PARTIAL SENSITIVITY FUNCTIONS

Although all the filters under investigation are seventh order low pass, they were chosen to be as different from each other as possible, in type and source termination resistances. As mentioned earlier these filters are identical to those analyzed in Chapter V for sensitivity



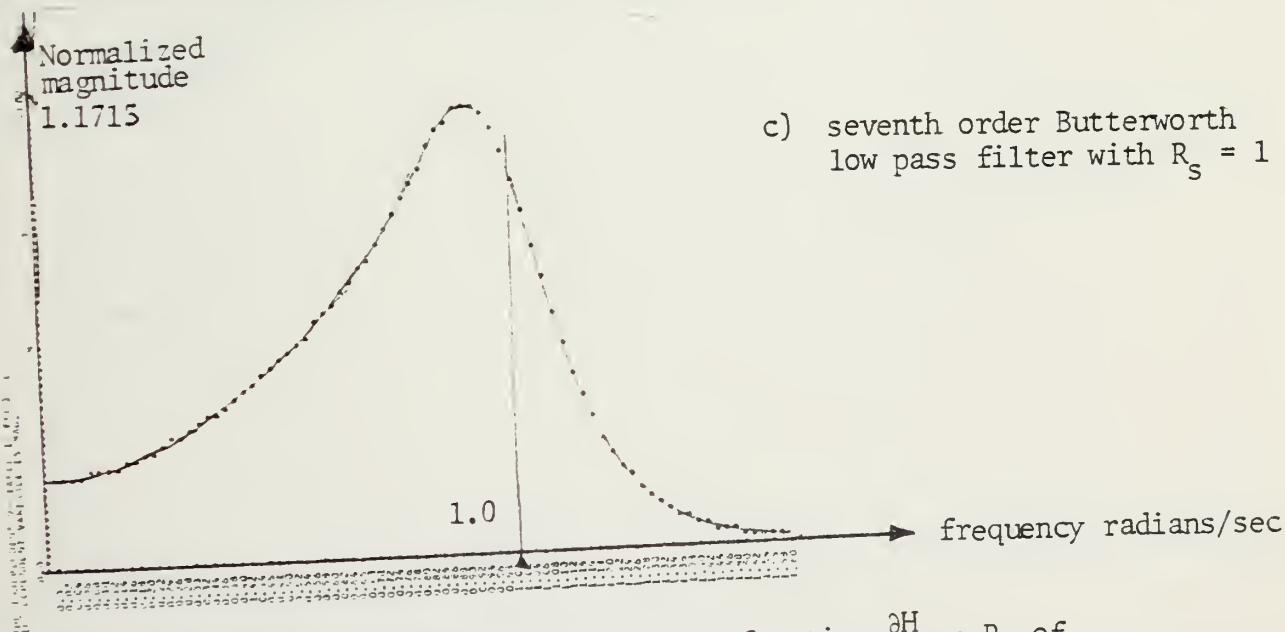
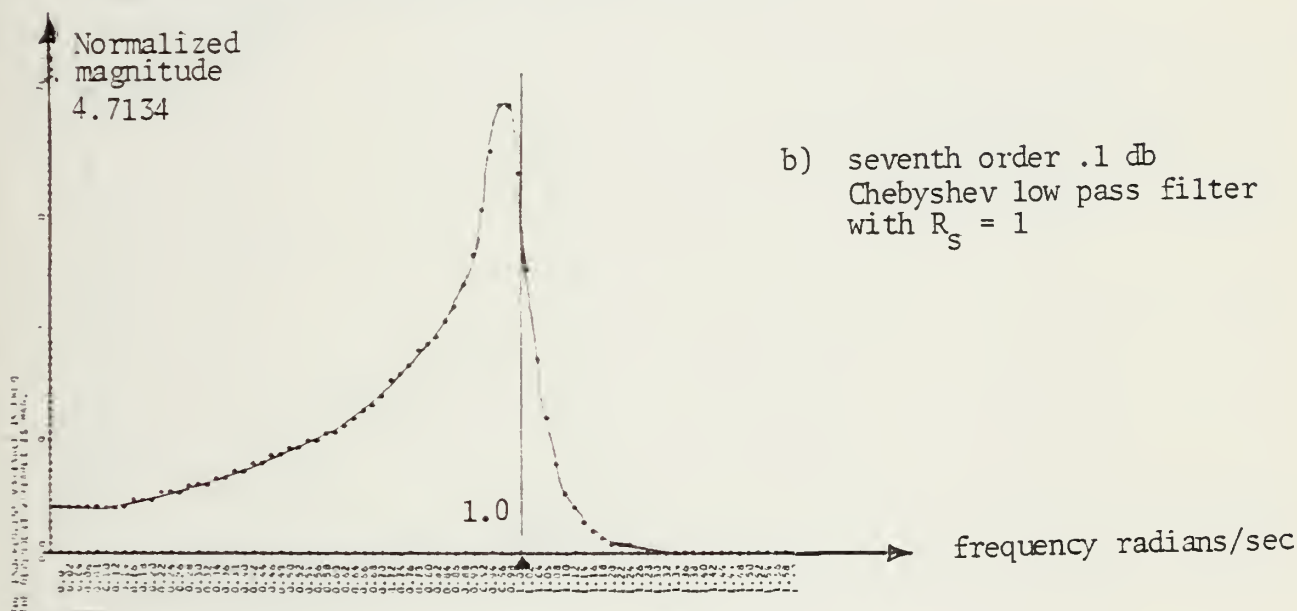
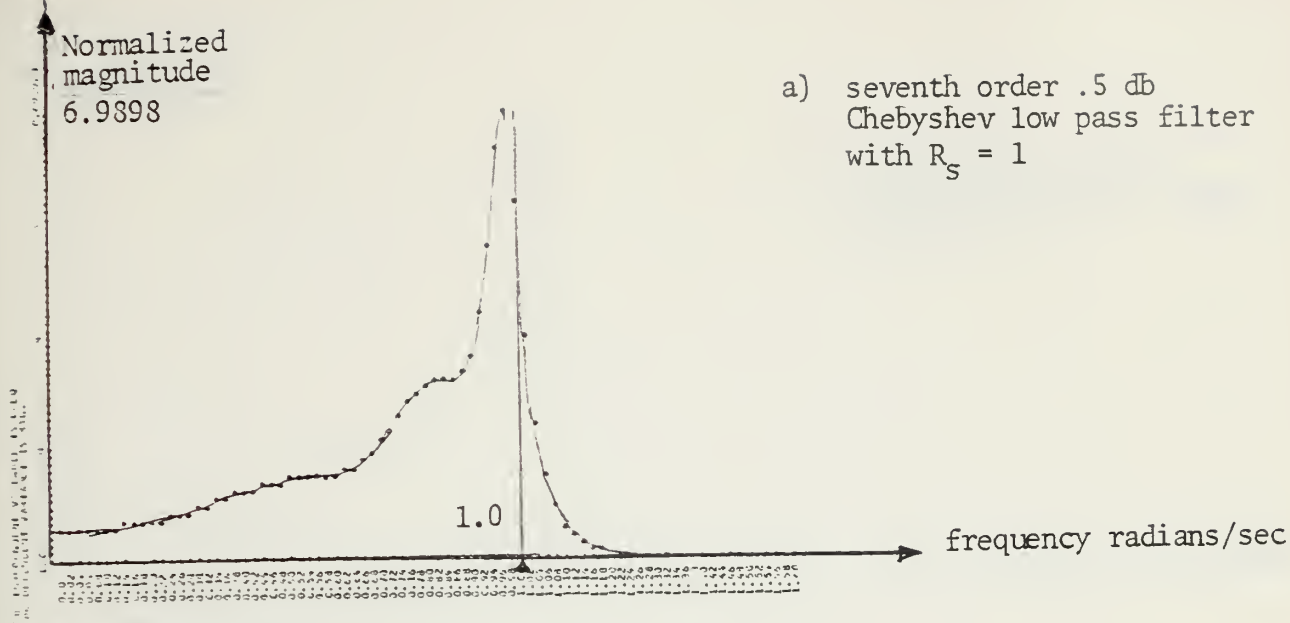


Fig. 6.5. The graphs of normalized sensitivity function  $\frac{\partial H}{\partial R_S} \cdot R_S$  of various simple wave digital filters.





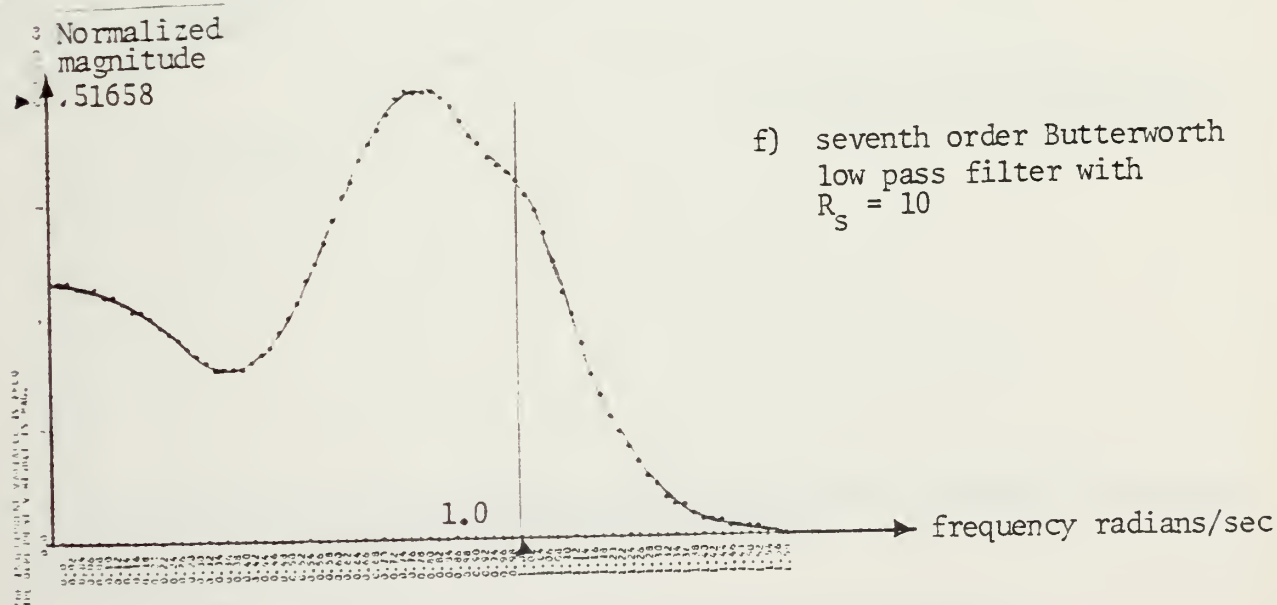
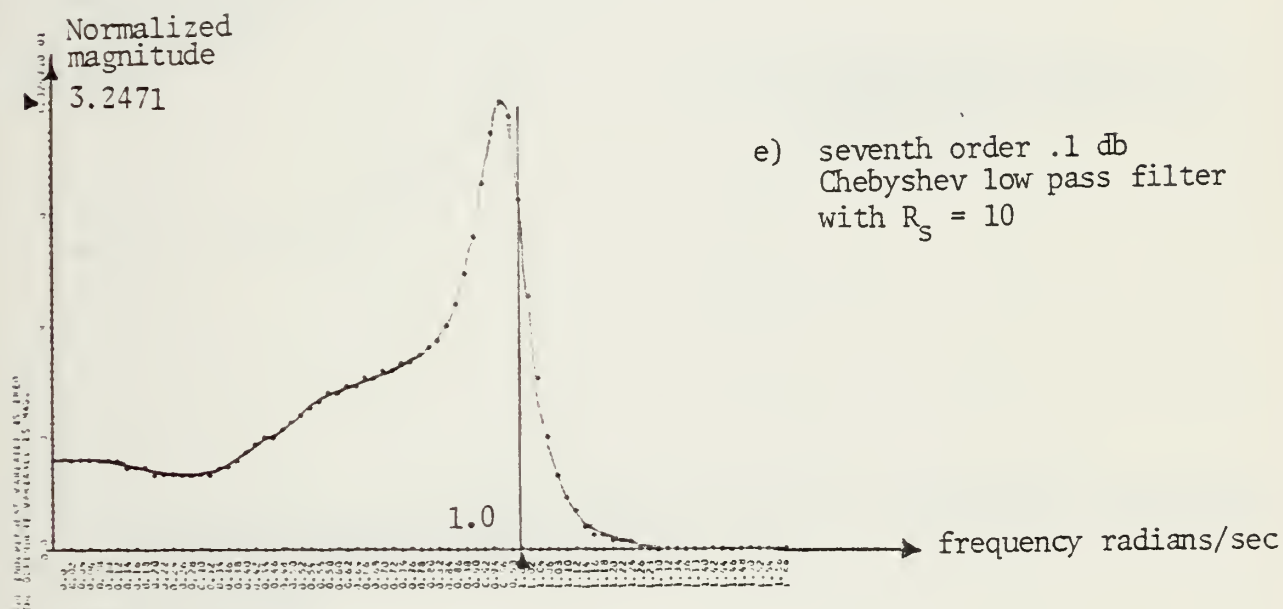
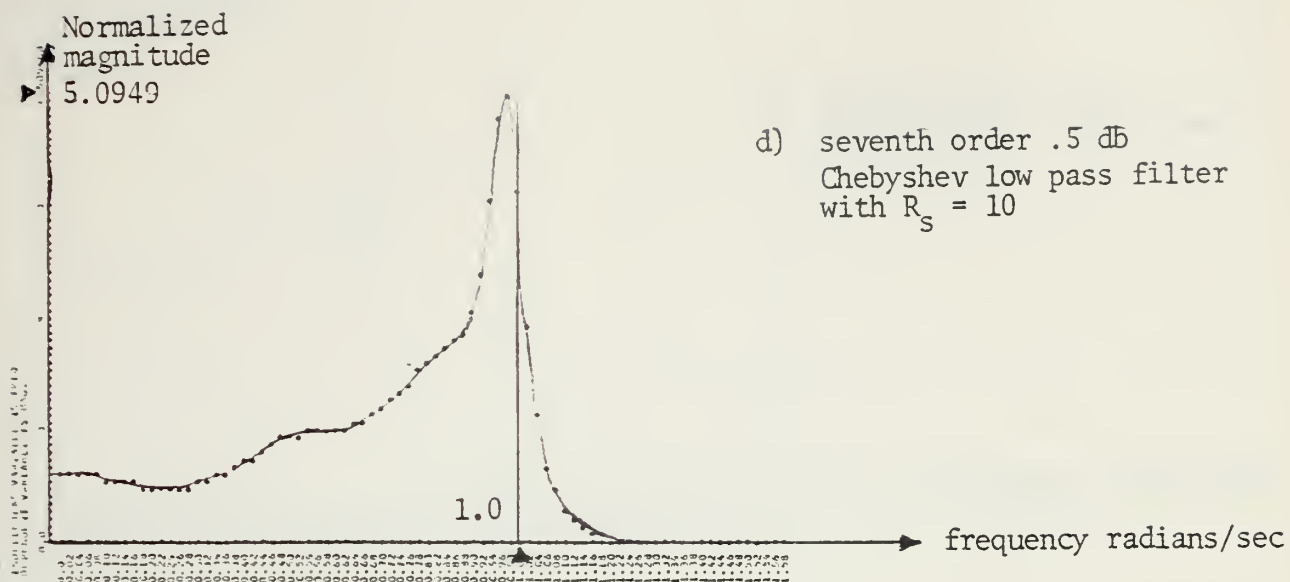


Fig. 6.5. Continued



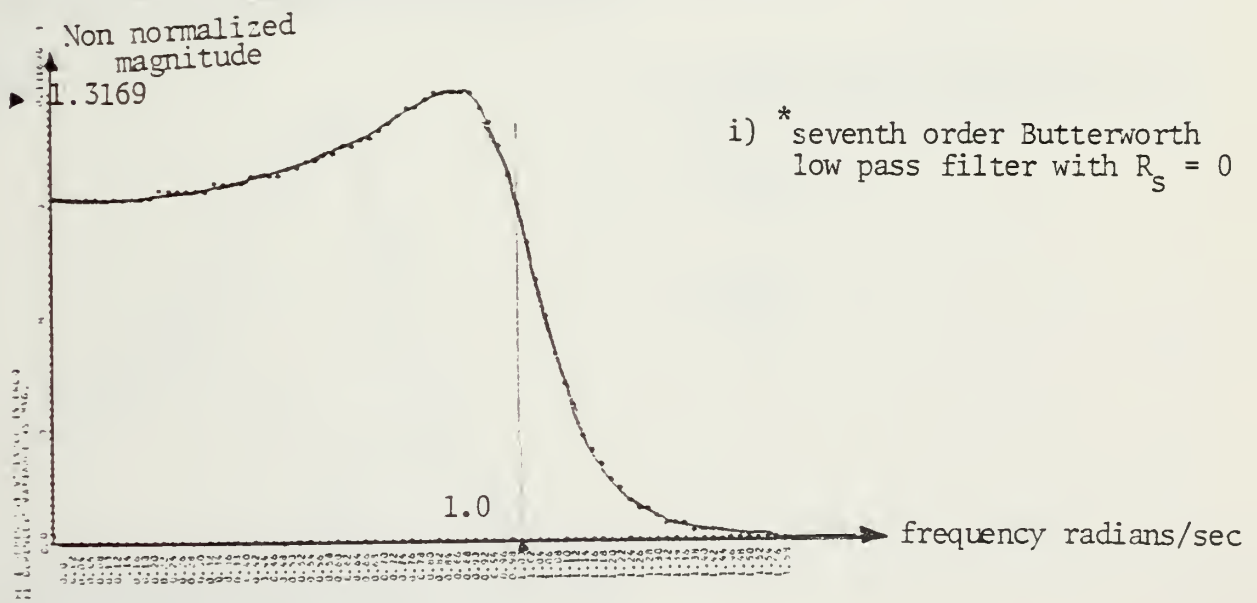
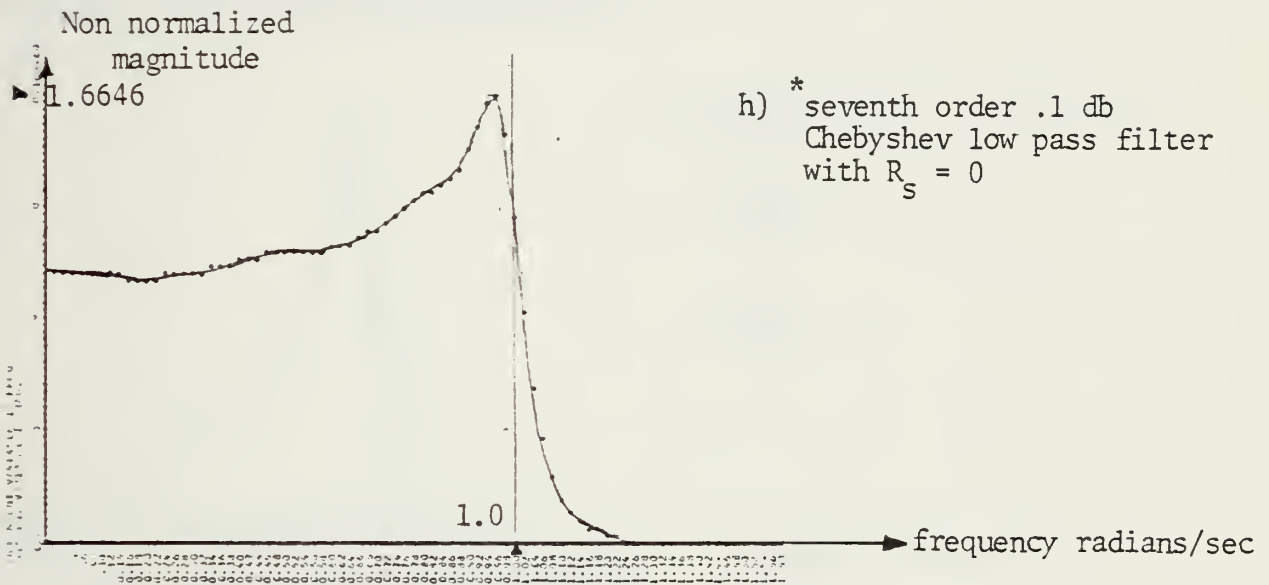
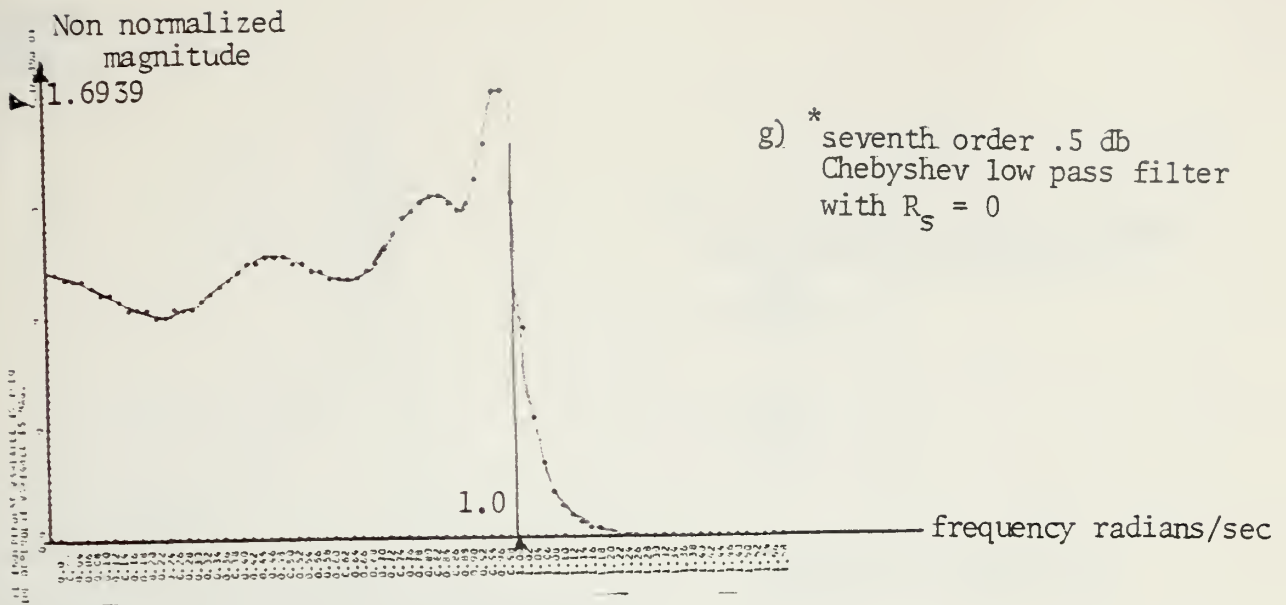


Fig. 6.5. Continued

\*Note that for the special case of  $R_S = 0$ , non normalized sensitivity function  $\frac{\partial H}{\partial R_S}$  are plotted.



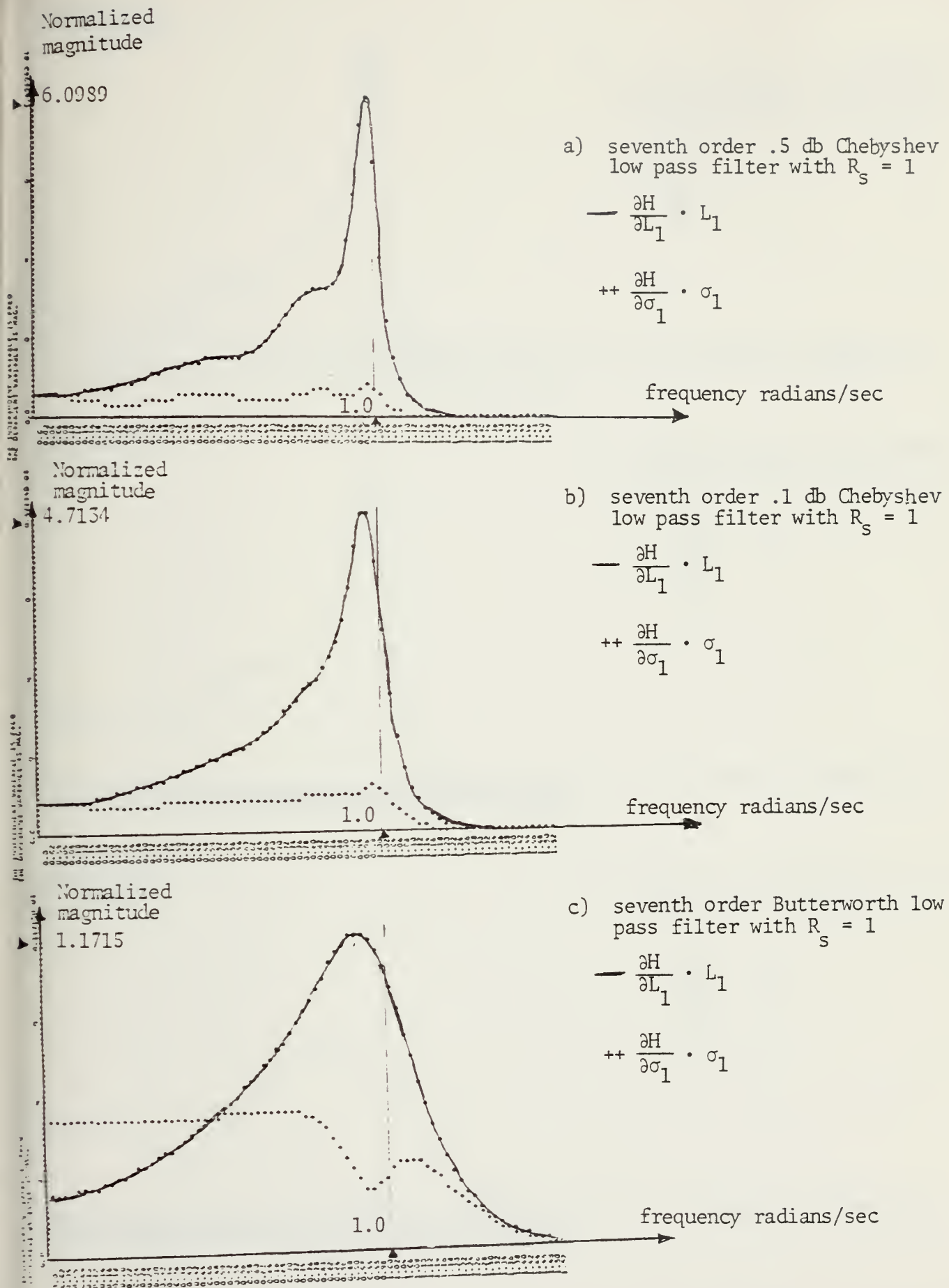


Fig. 6.6. The graphs of normalized sensitivity functions  $\frac{\partial H}{\partial L_1} \cdot L_1$  and  $\frac{\partial H}{\partial \sigma_1} \cdot \sigma_1$  of various simple wave digital filters.



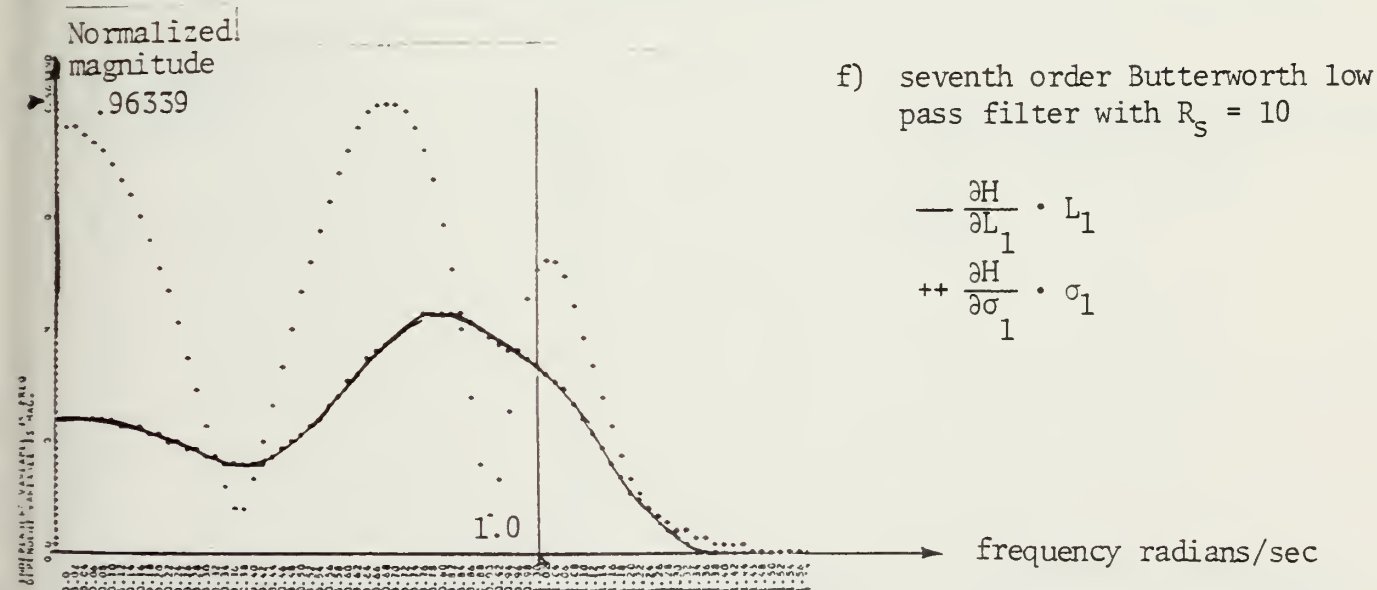
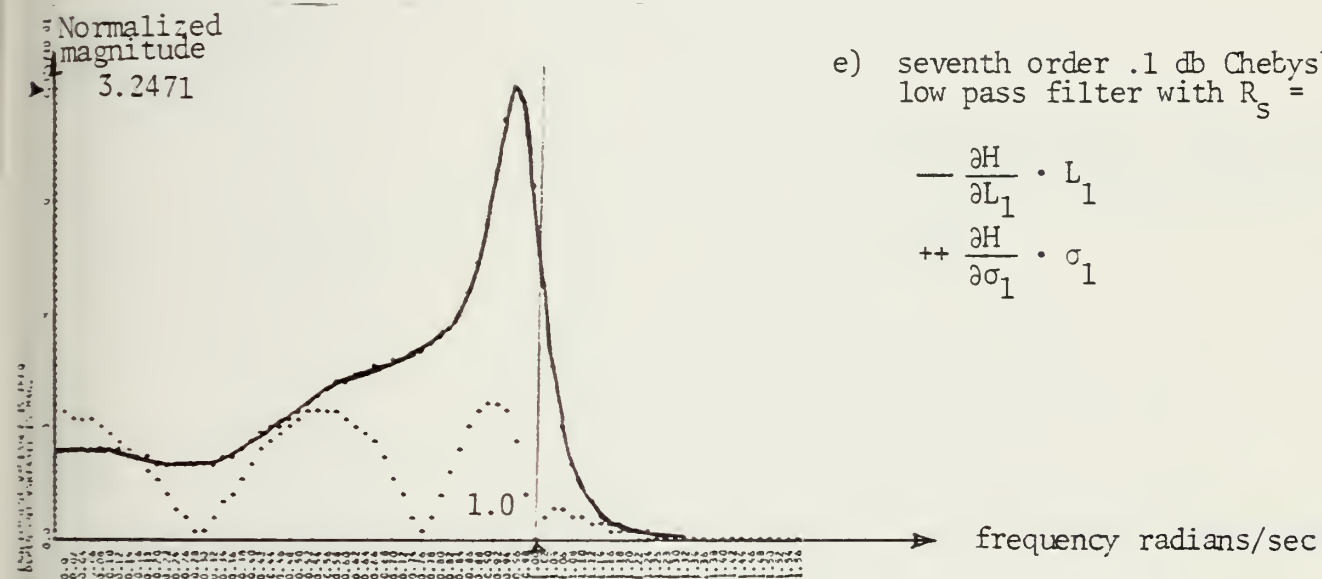
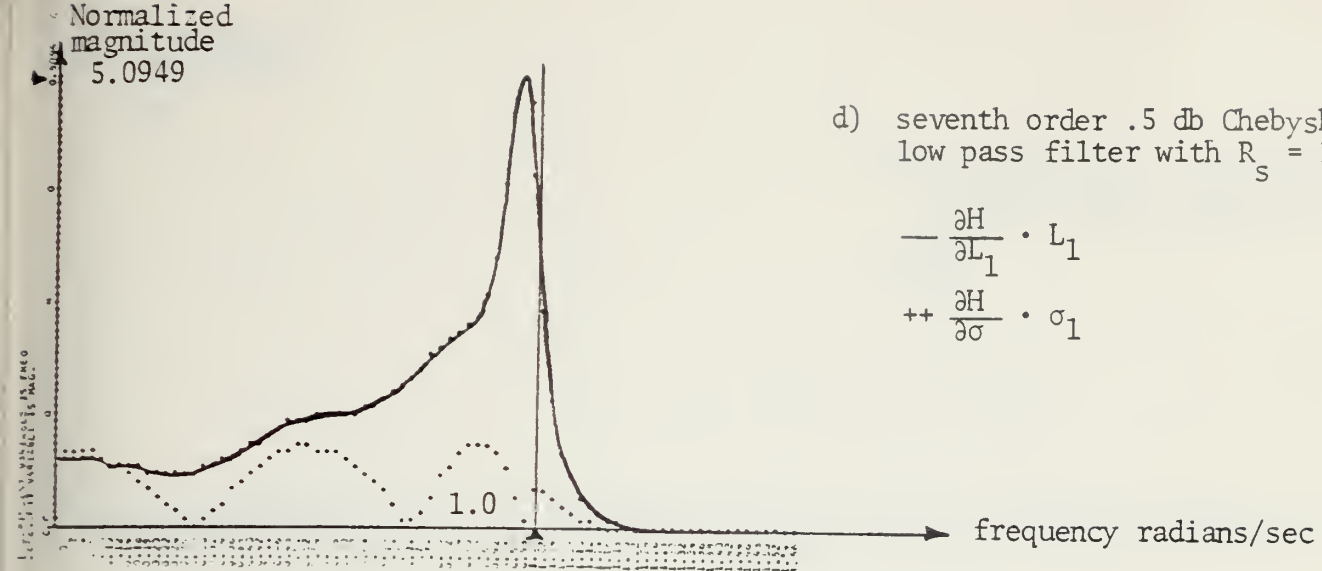


Fig. 6.6. Continued





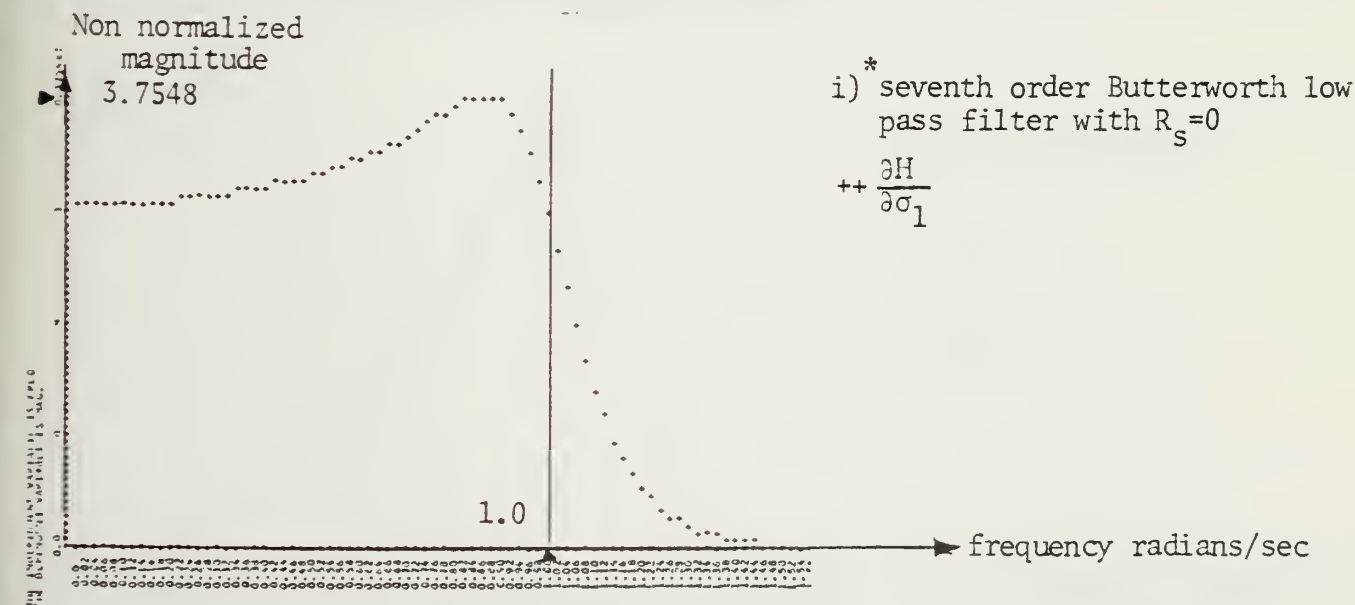
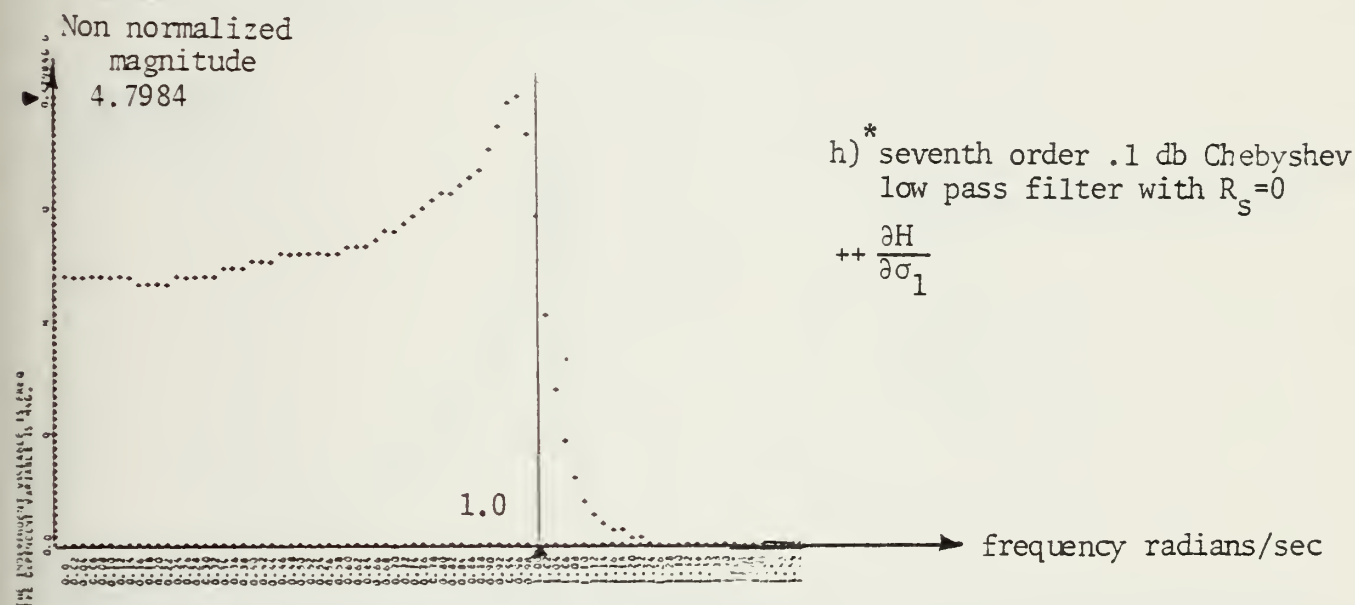
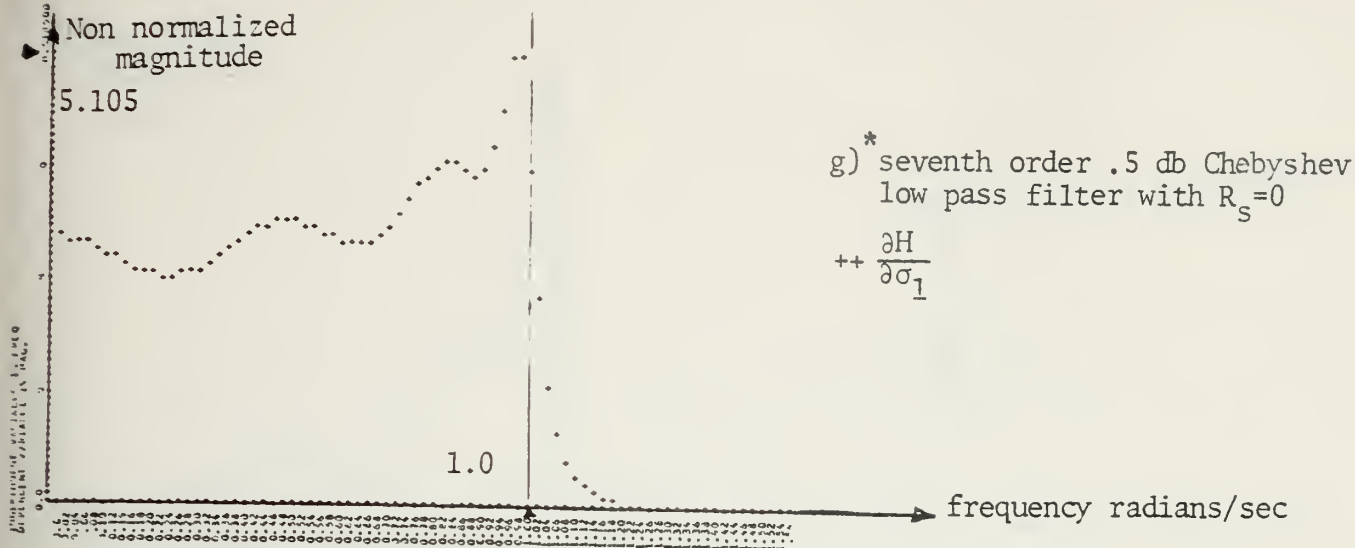


Fig. 6.6. Continued \*For the special case of  $R_s=0$  non normalized curves are plotted.



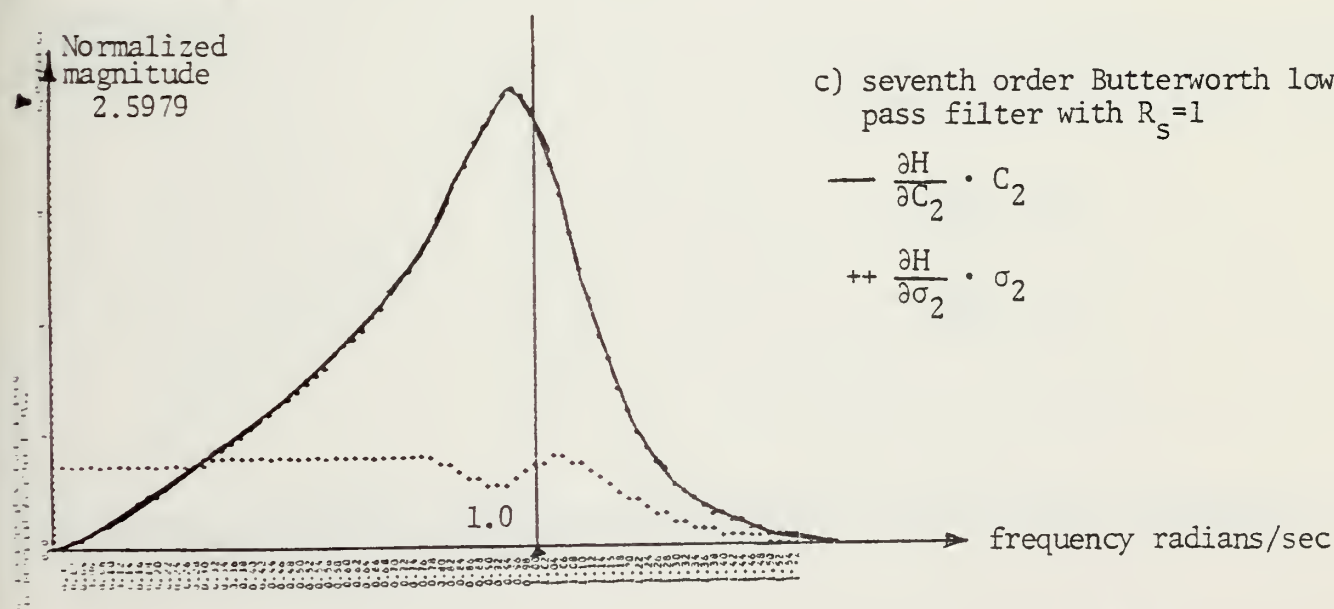
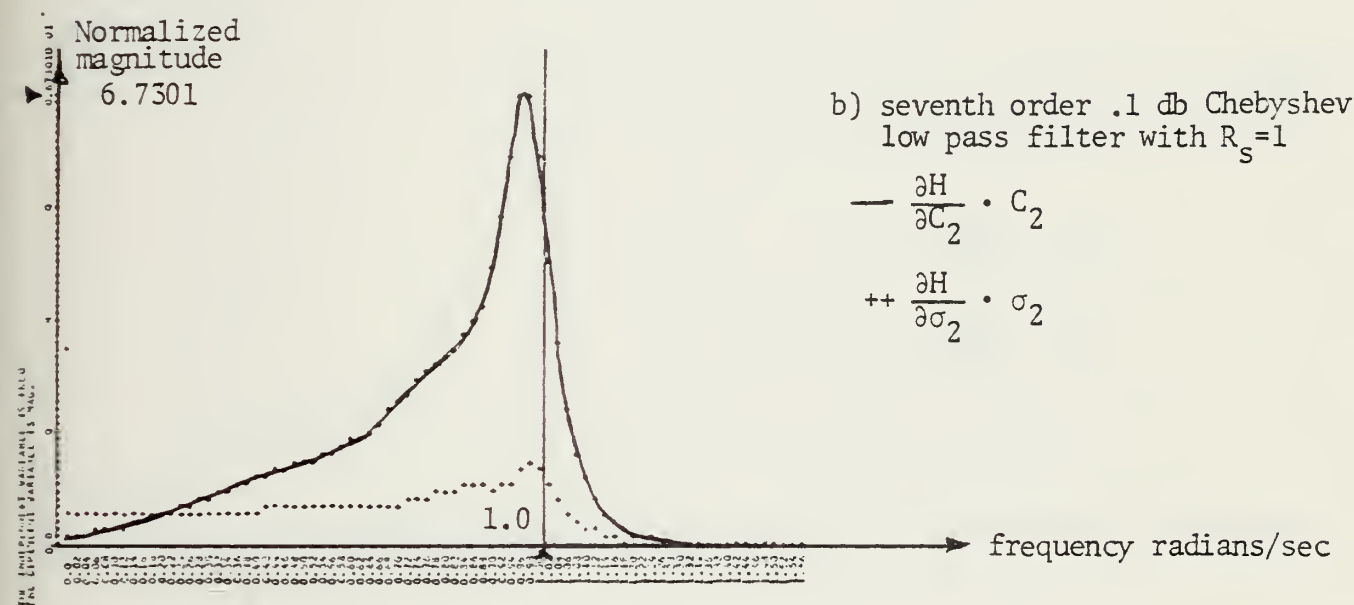
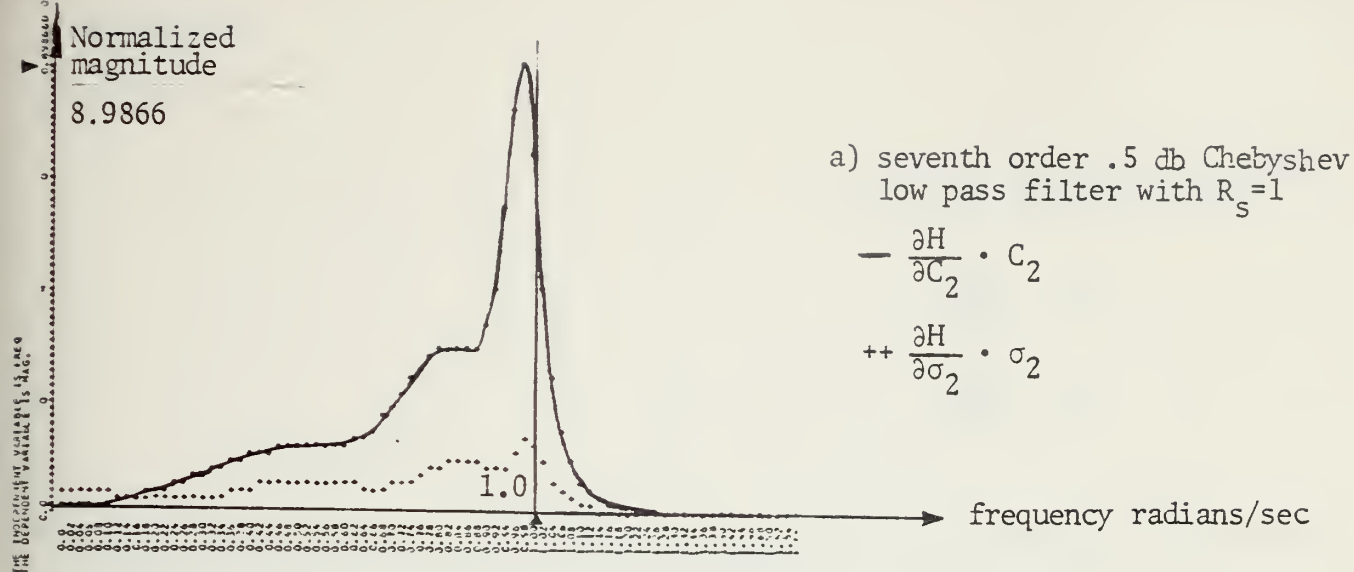


Fig. 6.7. The graphs of normalized sensitivity function,  $\frac{\partial H}{\partial C_2} \cdot C_2$  and  $\frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$  of various simple wave digital filters.



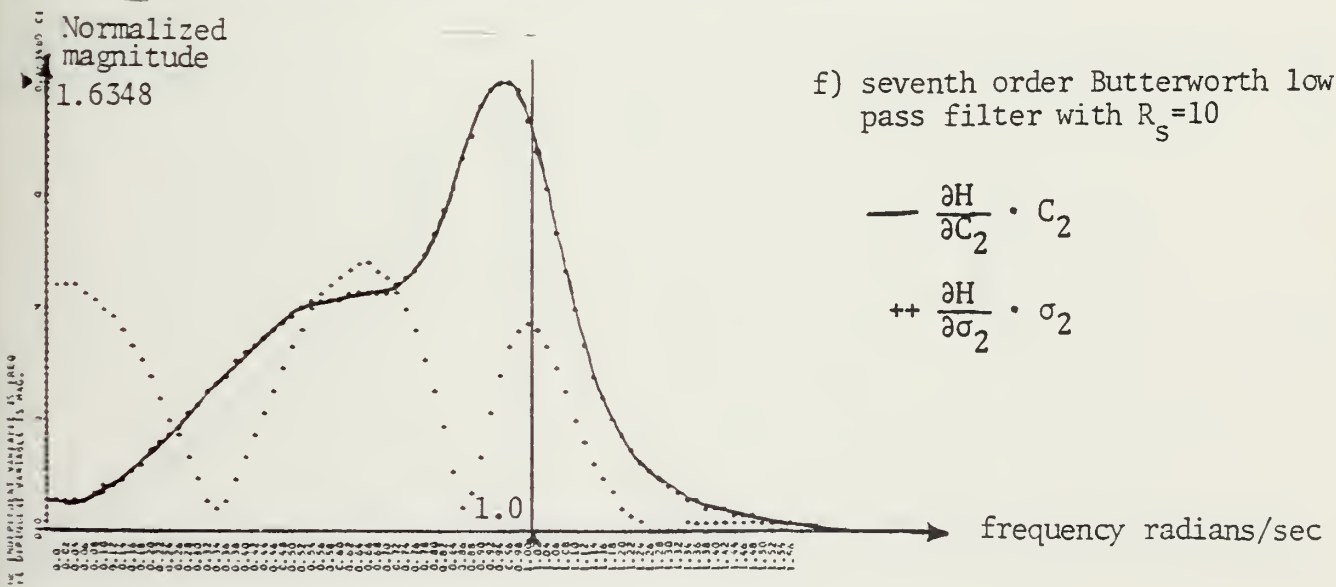
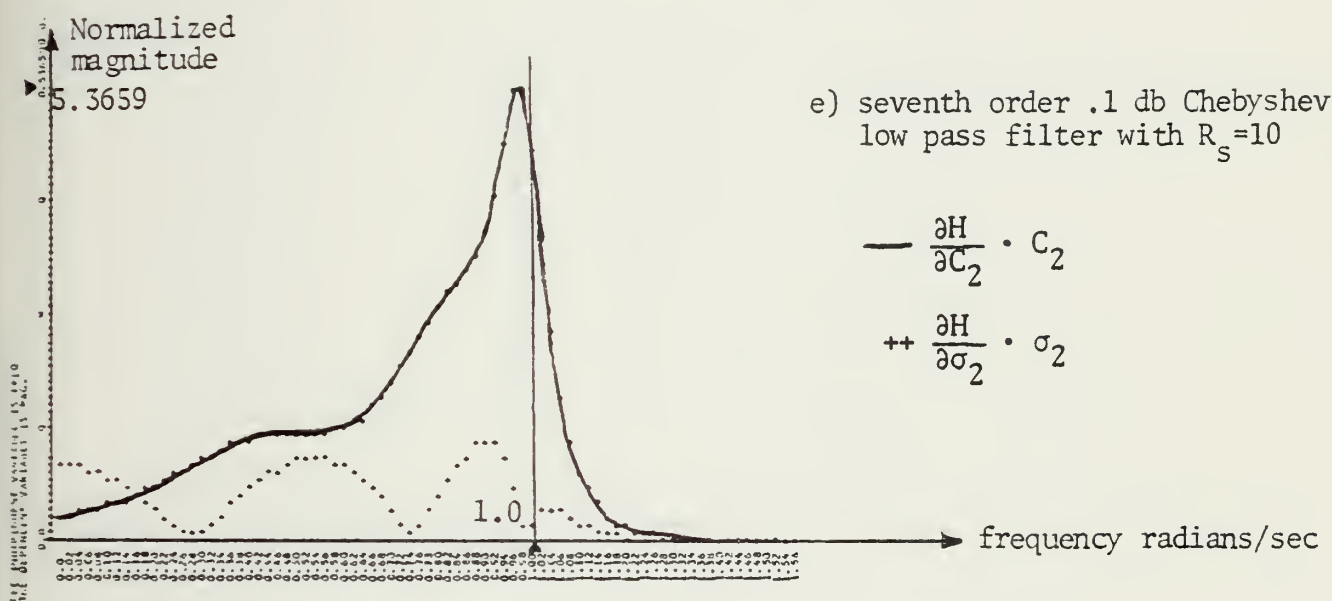
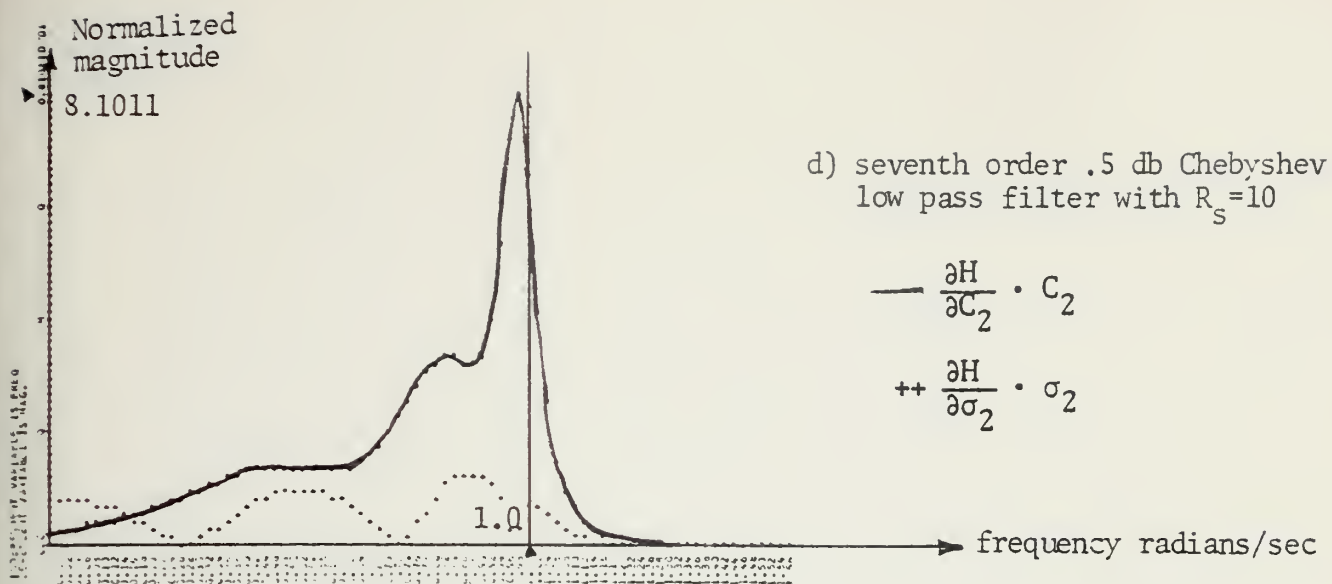
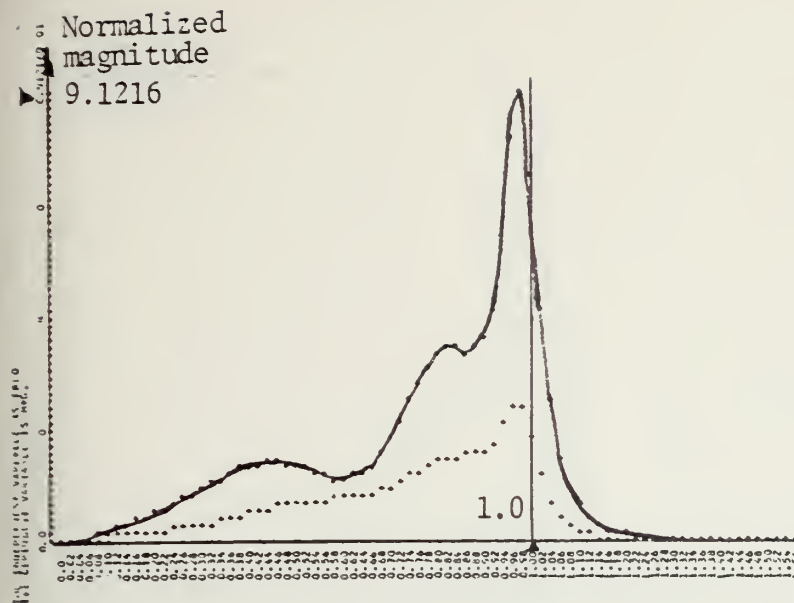


Fig. 6.7. Continued

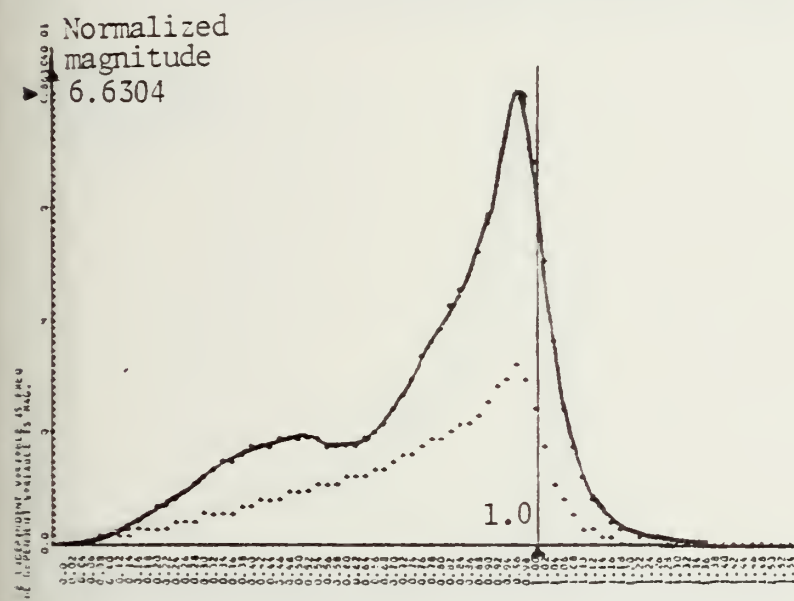




g) seventh order .5 db Chebyshev  
low pass filter with  $R_s=0$

$$- \frac{\partial H}{\partial C_2} \cdot C_2$$

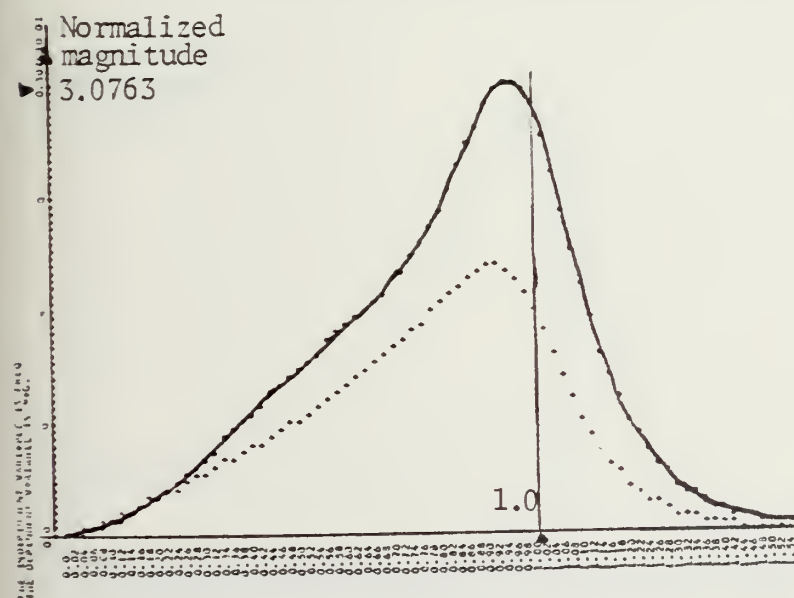
$$++ \frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$$



h) seventh order .1 db Chebyshev  
low pass filter with  $R_s=0$

$$- \frac{\partial H}{\partial C_2} \cdot C_2$$

$$++ \frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$$



i) seventh order Butterworth low  
pass filter with  $R_s=0$

$$- \frac{\partial H}{\partial C_2} \cdot C_2$$

$$++ \frac{\partial H}{\partial \sigma_2} \cdot \sigma_2$$

Fig. 6.7. Continued





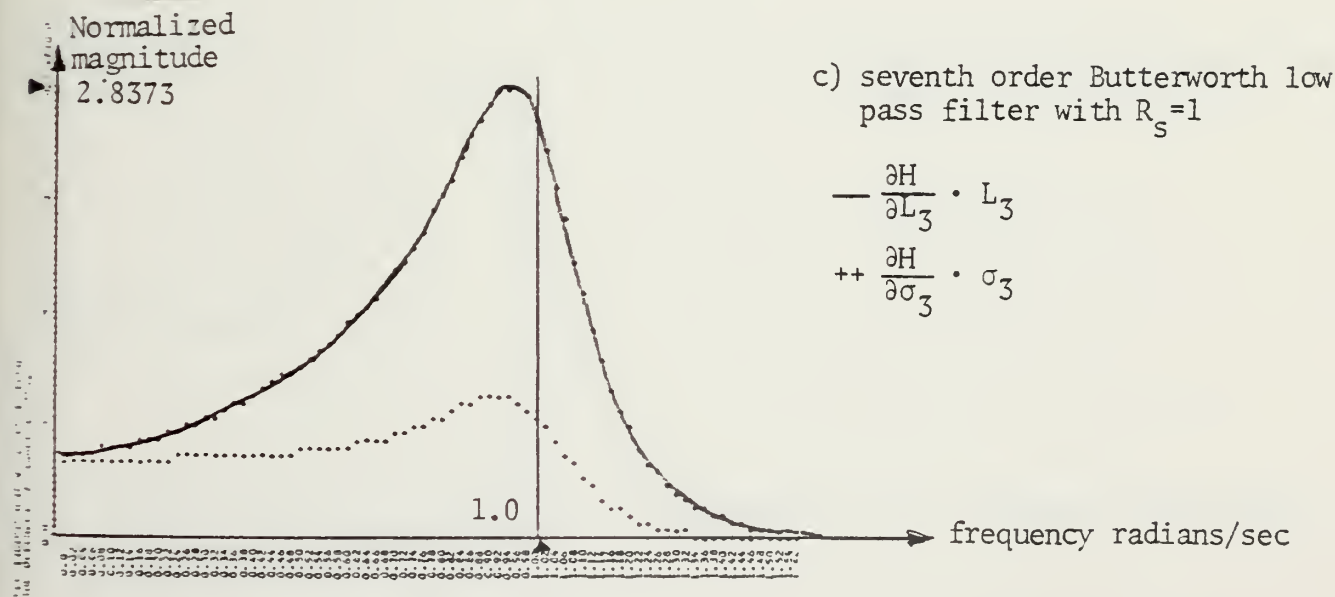
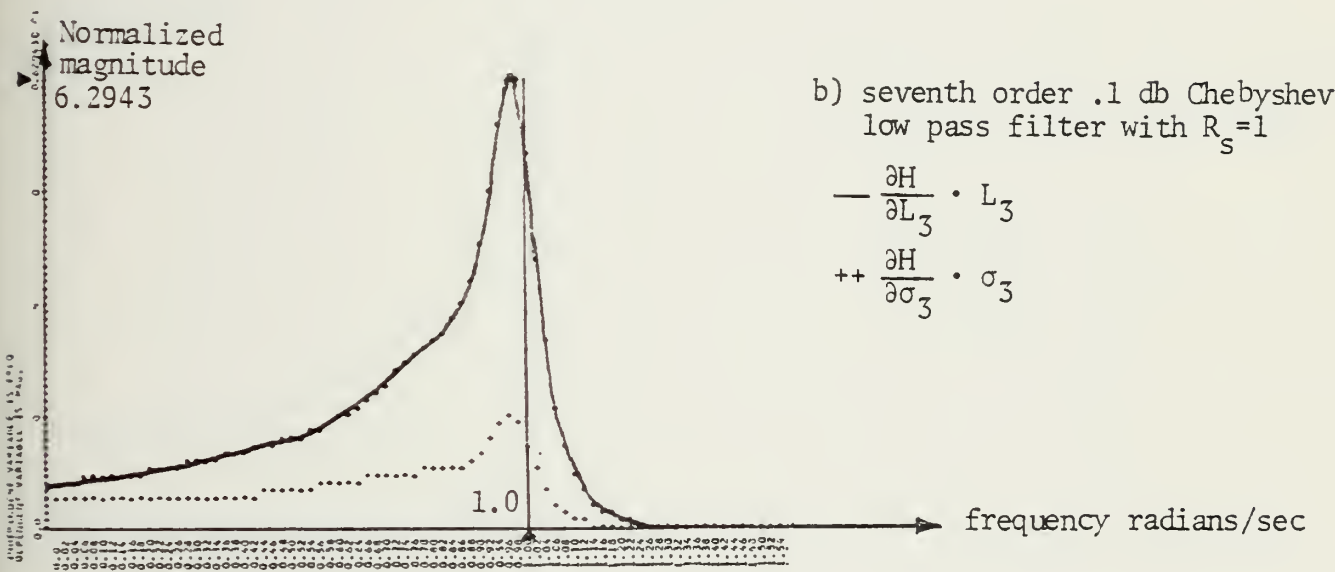
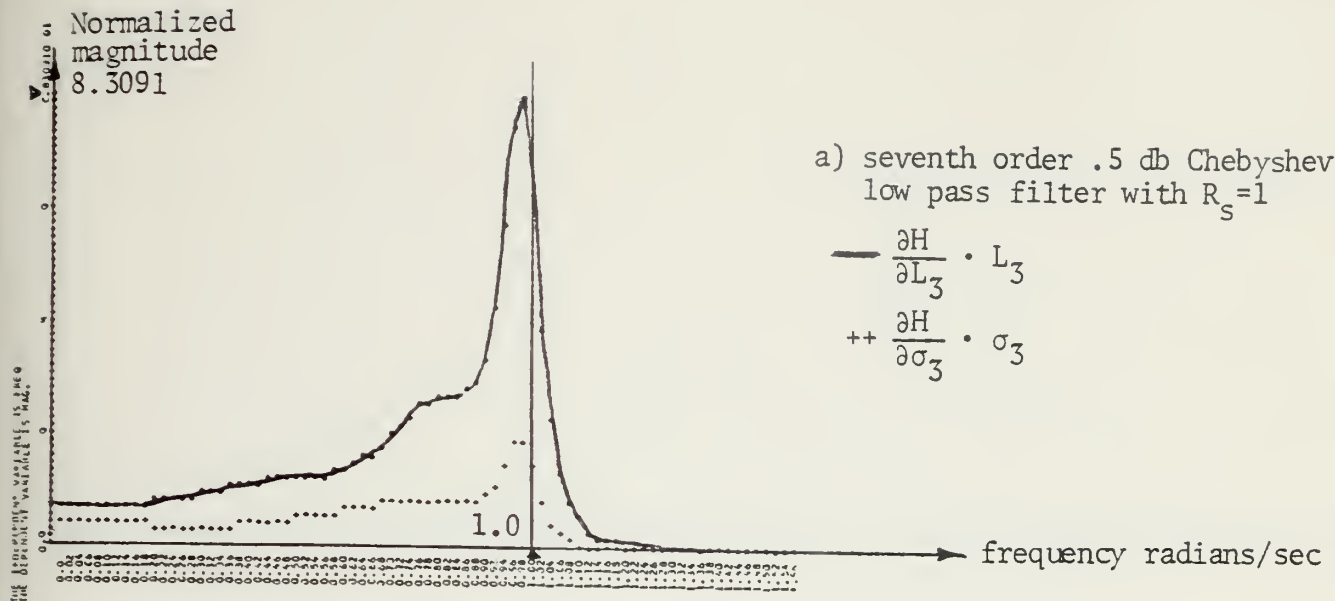


Fig. 6.8. The graphs of normalized sensitivity function,  $\frac{\partial H}{\partial L_3} \cdot L_3$  and  $\frac{\partial H}{\partial \sigma_3} \cdot \sigma_3$  of various simple wave digital filters.



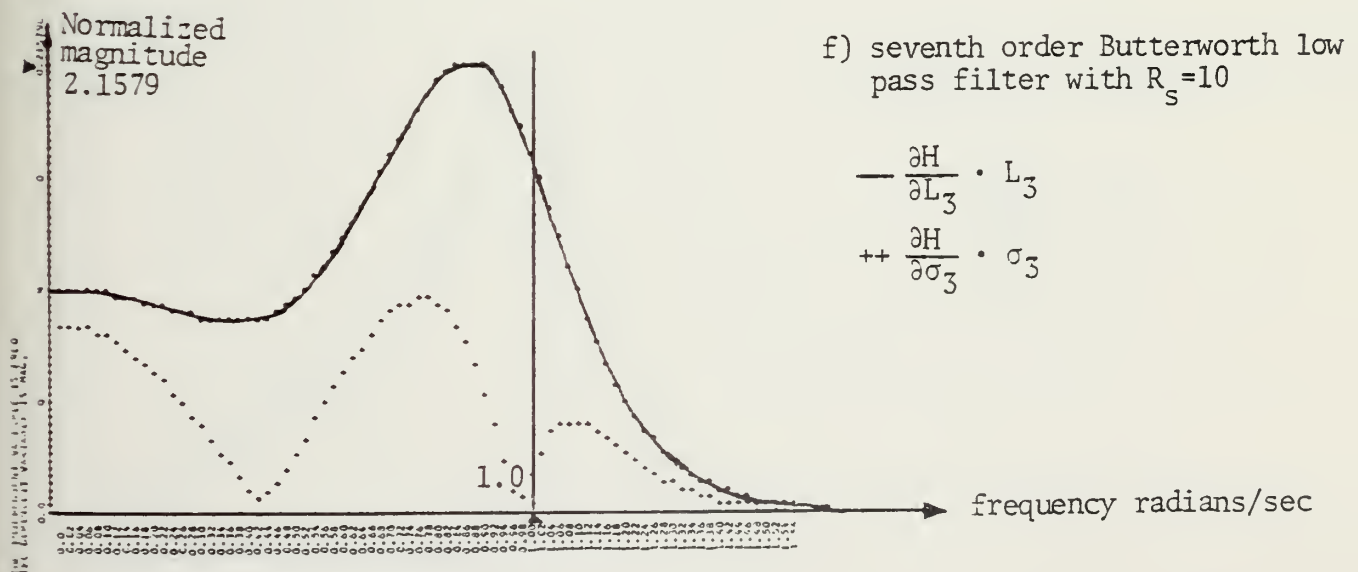
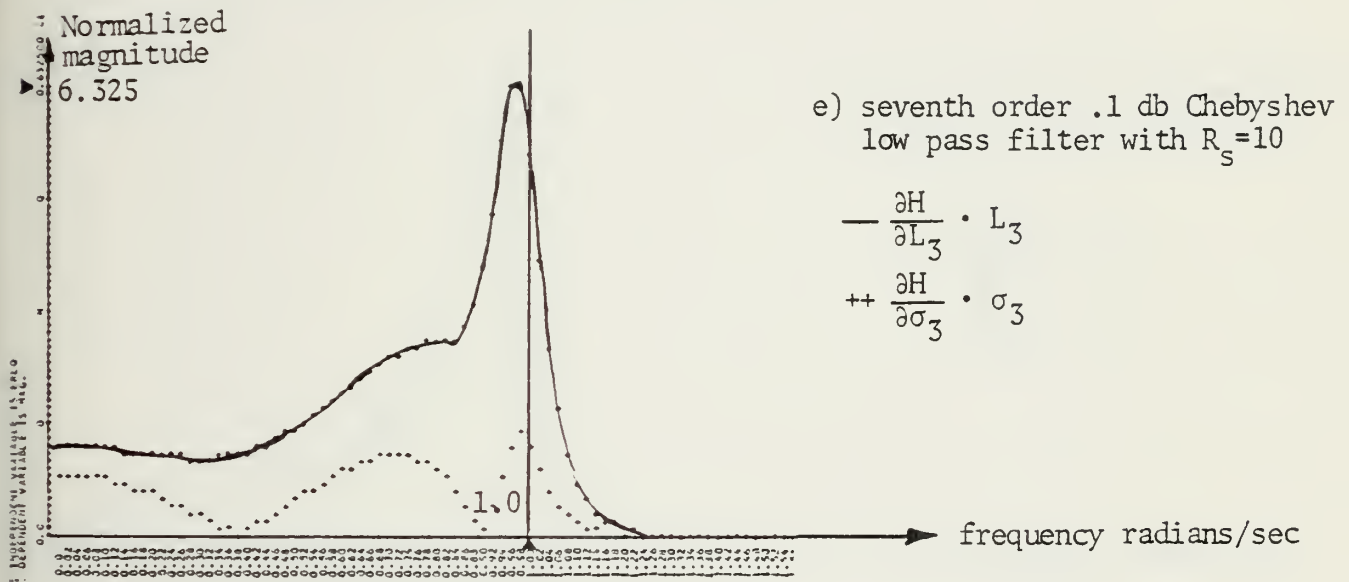
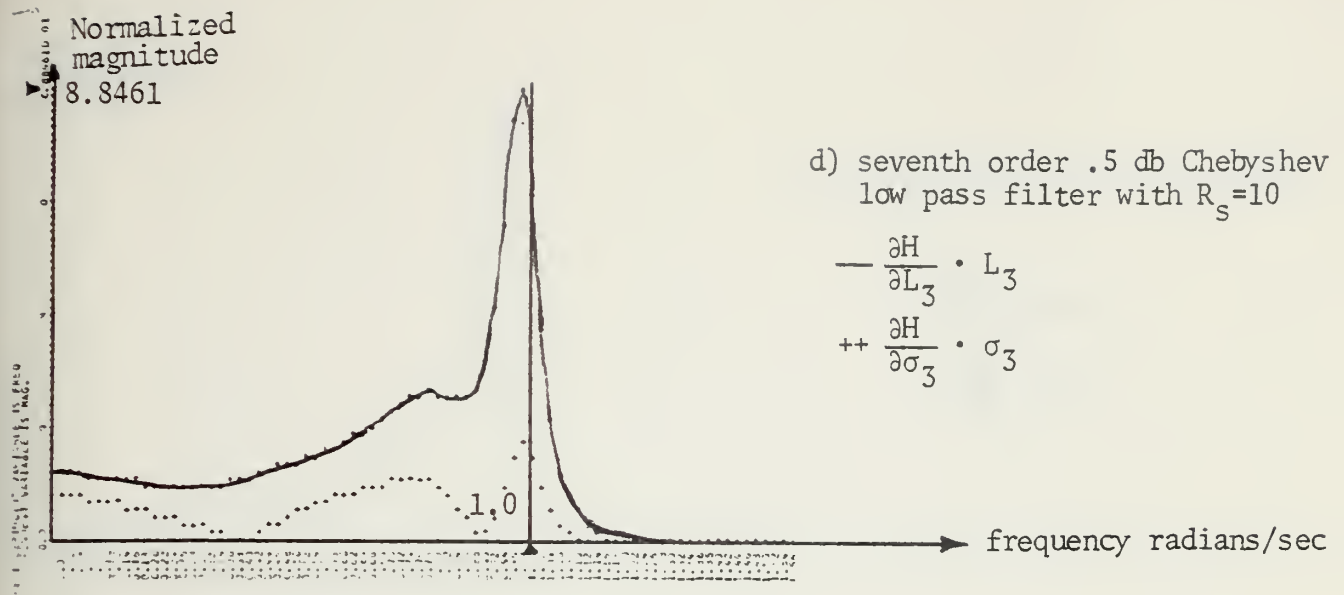
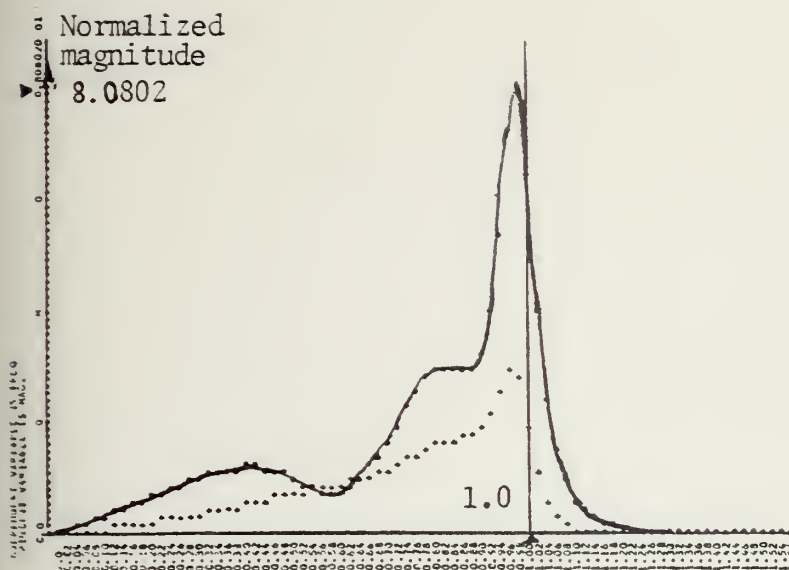


Fig. 6.8. Continued

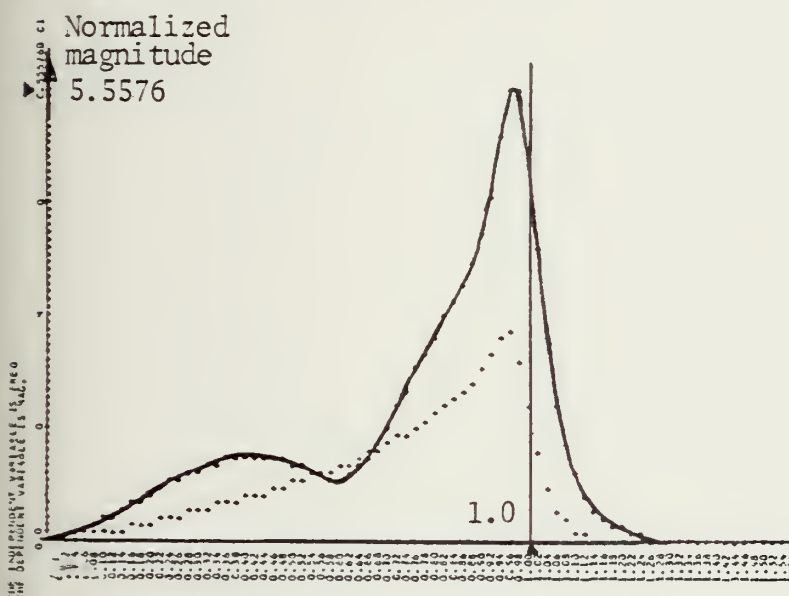




g) seventh order .5 db Chebyshev  
low pass filter with  $R_s=0$

$$\text{---} \frac{\partial H}{\partial L_3} \cdot L_3$$

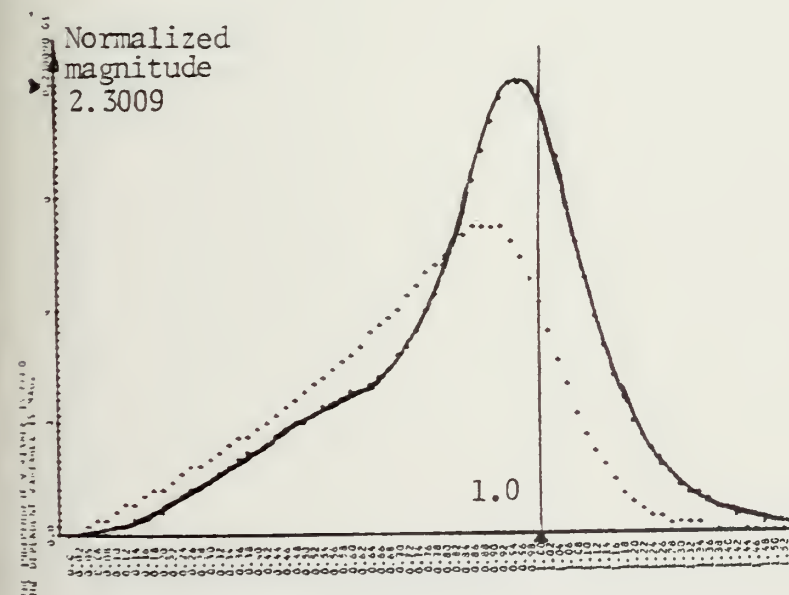
$$++ \frac{\partial H}{\partial \sigma_3} \cdot \sigma_3$$



h) seventh order .1 db Chebyshev  
low pass filter with  $R_s=0$

$$\text{---} \frac{\partial H}{\partial L_3} \cdot L_3$$

$$++ \frac{\partial H}{\partial \sigma_3} \cdot \sigma_3$$



i) seventh order Butterworth low  
pass filter with  $R_s=0$

$$\text{---} \frac{\partial H}{\partial L_3} \cdot L_3$$

$$++ \frac{\partial H}{\partial \sigma_3} \cdot \sigma_3$$

Fig. 6.8. Continued.



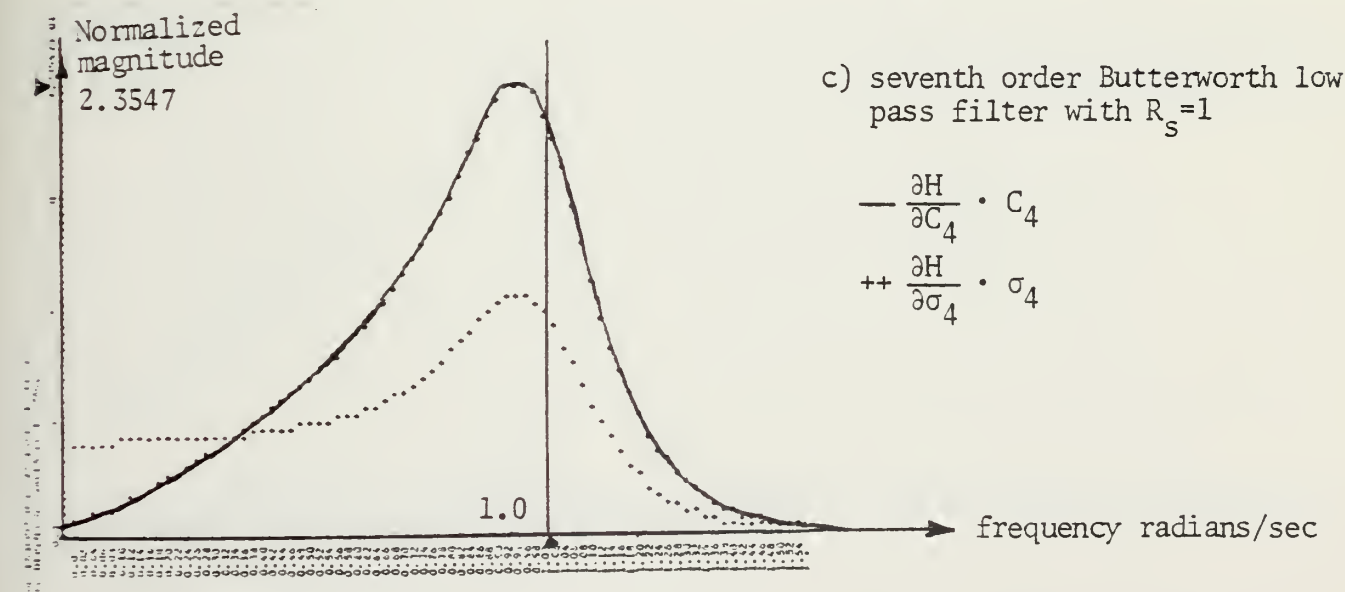
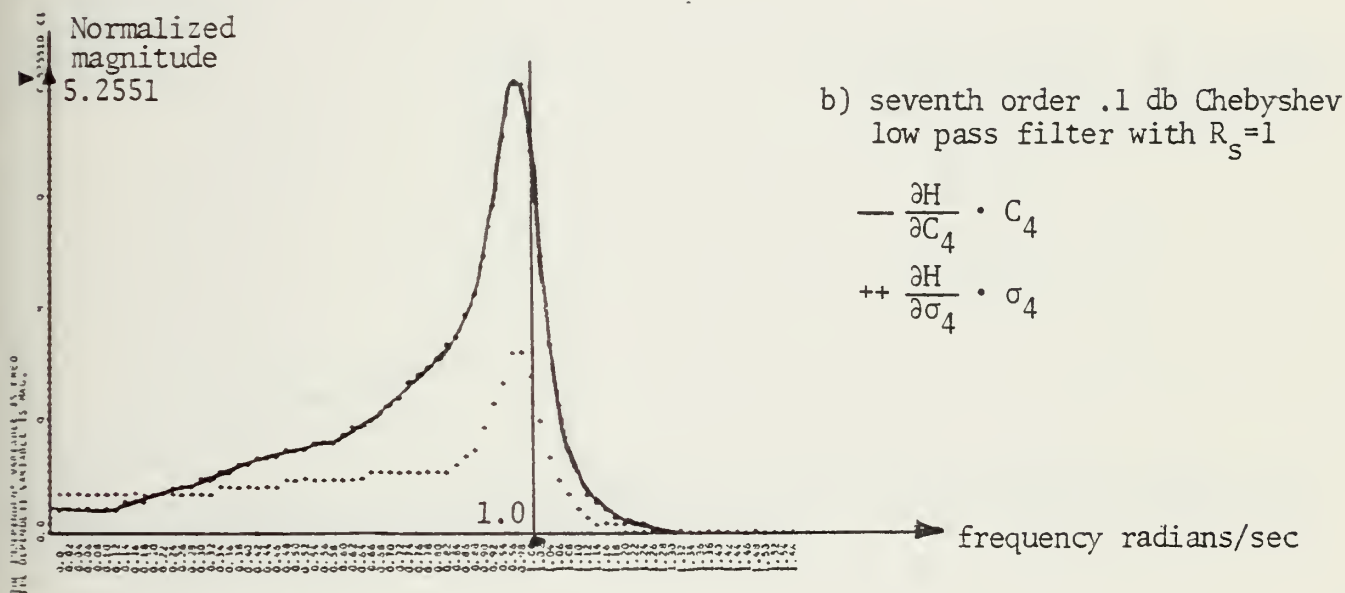
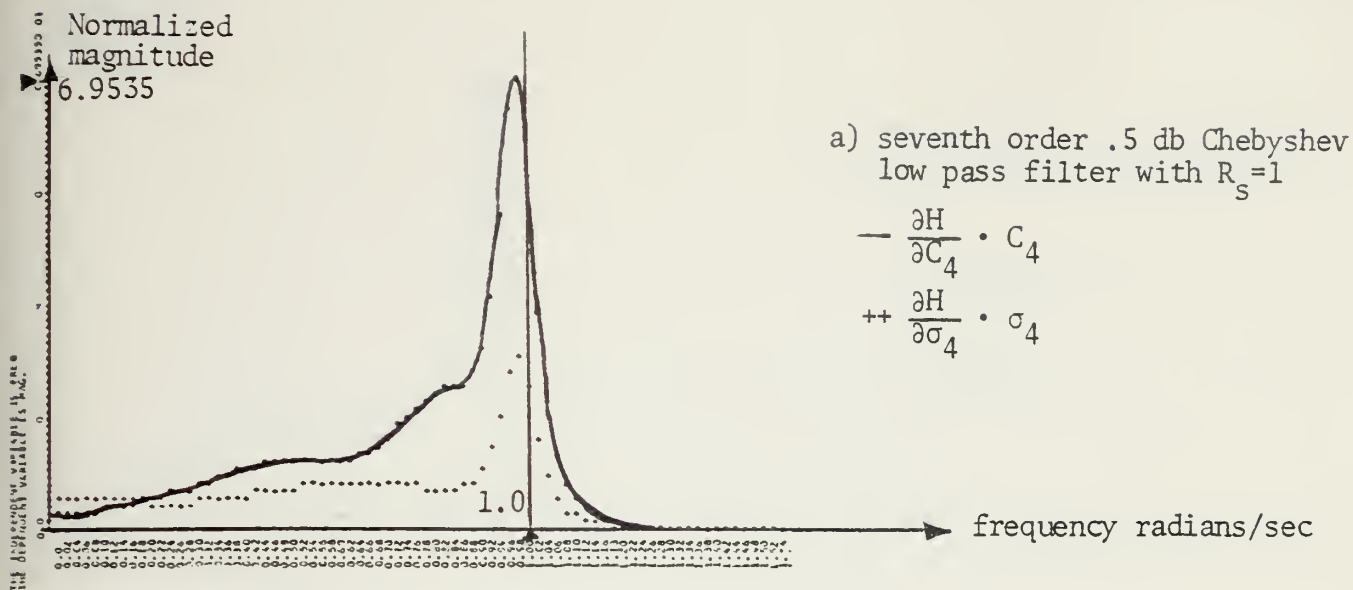


Fig. 6.9. The graphs of normalized sensitivity function,  $\frac{\partial H}{\partial C_4} \cdot C_4$  and  $\frac{\partial H}{\partial \sigma_4} \cdot \sigma_4$  of various simple wave digital filters.





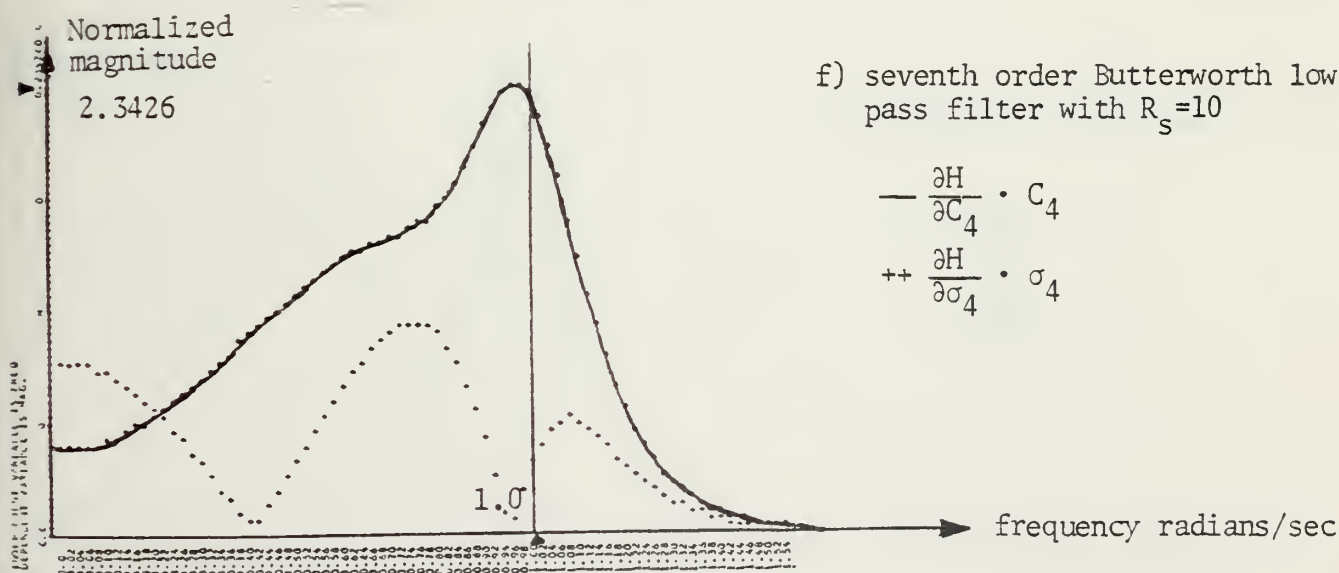
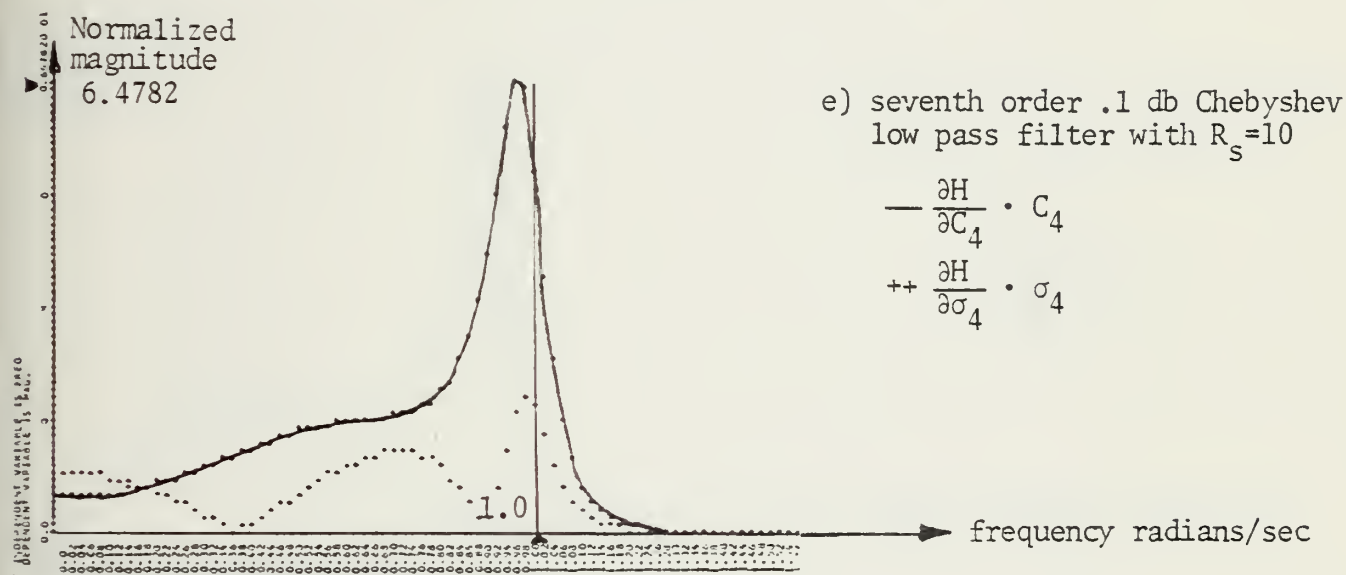
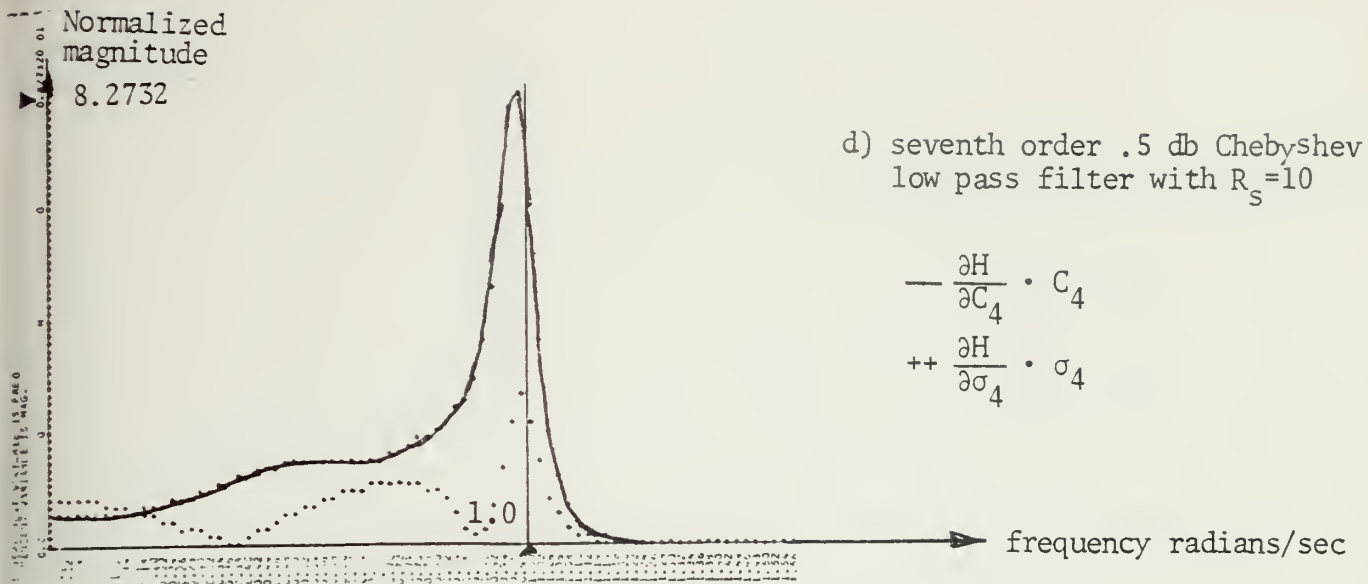
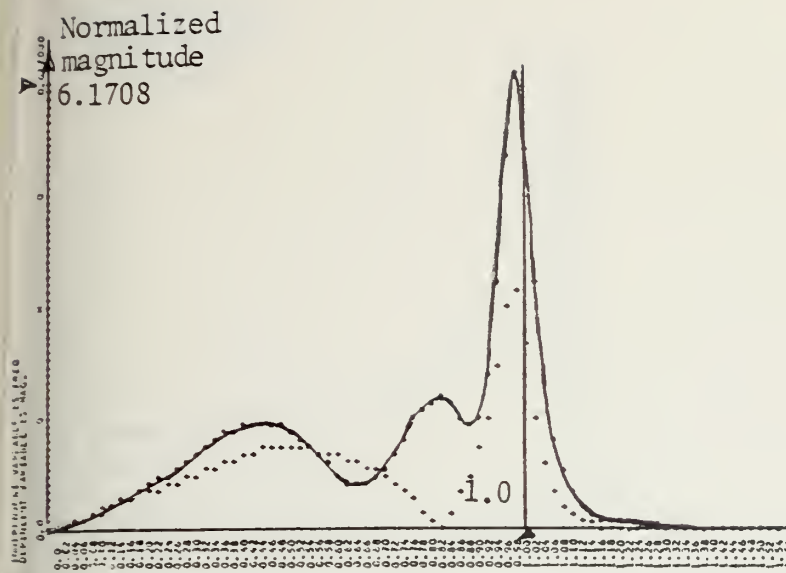


Fig. 6.9. Continued.

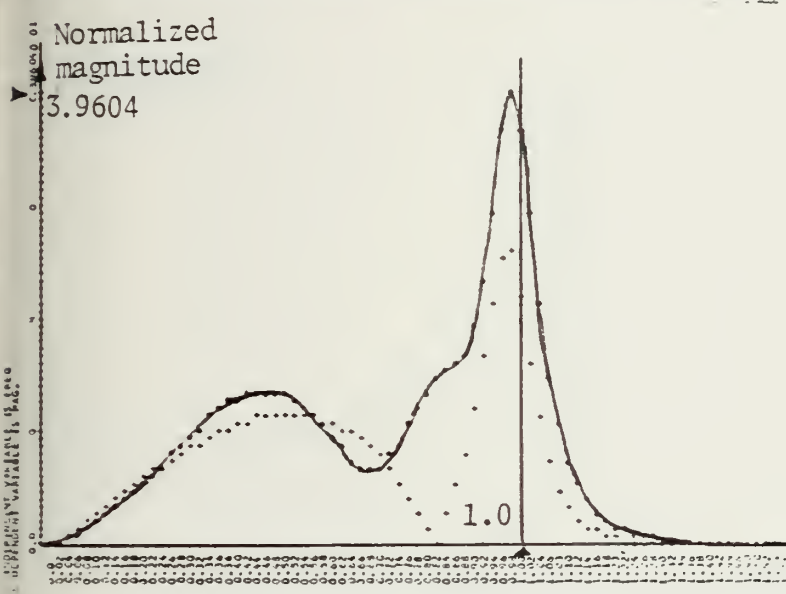




g) seventh order .5 db Chebyshev low pass filter with  $R_s=0$

$$\text{--- } \frac{\partial H}{\partial C_4} \cdot C_4$$

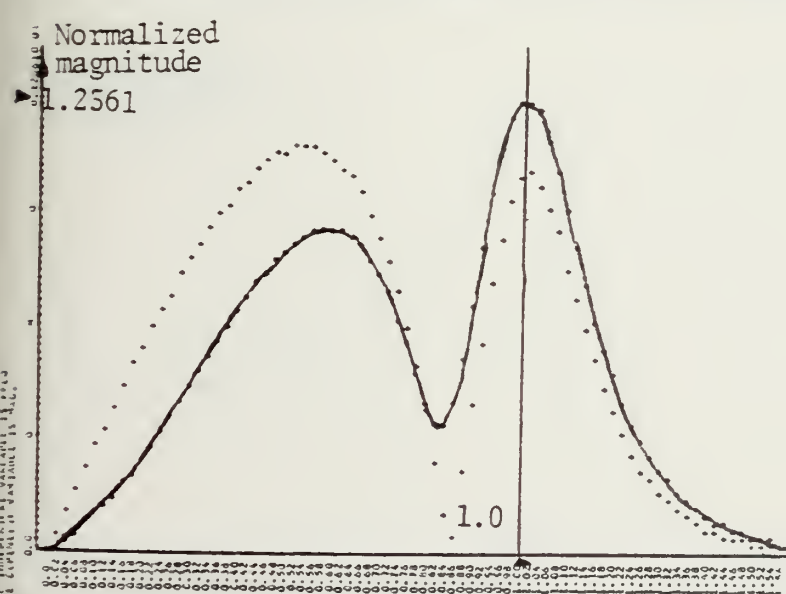
$$\text{++ } \frac{\partial H}{\partial \sigma_4} \cdot \sigma_4$$



h) seventh order .1 db Chebyshev low pass filter with  $R_s=0$

$$\text{--- } \frac{\partial H}{\partial C_4} \cdot C_4$$

$$\text{++ } \frac{\partial H}{\partial \sigma_4} \cdot \sigma_4$$



i) seventh order Butterworth low pass filter with  $R_s=0$

$$\text{--- } \frac{\partial H}{\partial C_4} \cdot C_4$$

$$\text{++ } \frac{\partial H}{\partial \sigma_4} \cdot \sigma_4$$

Fig. 6.9. Continued.



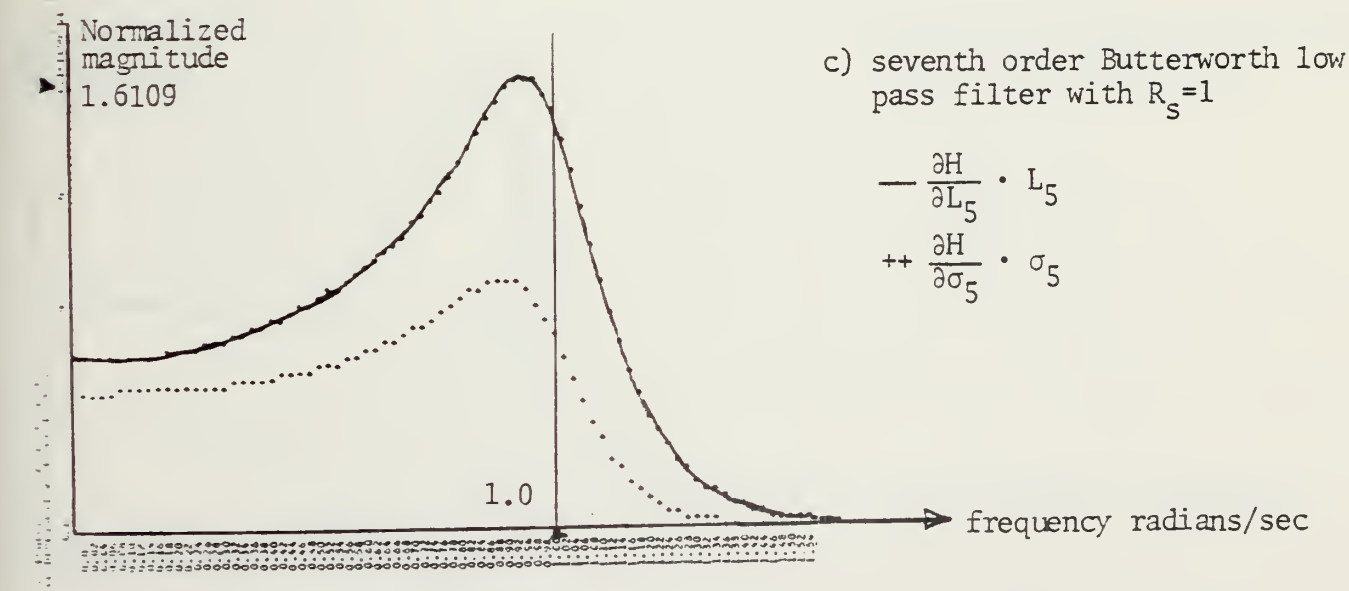
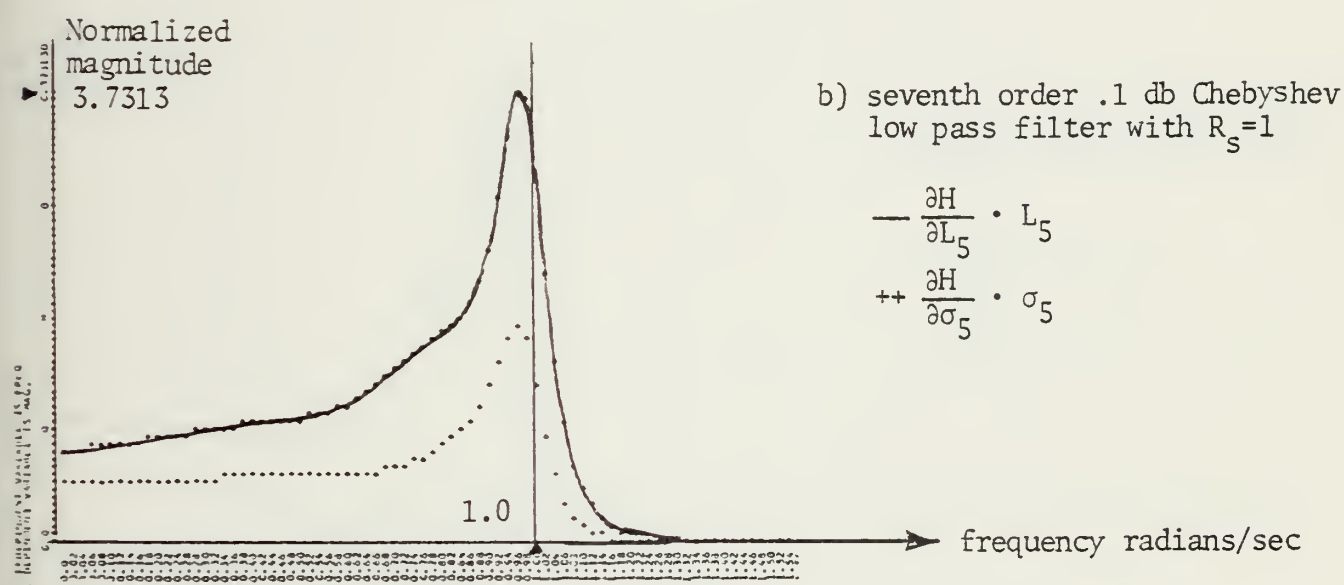
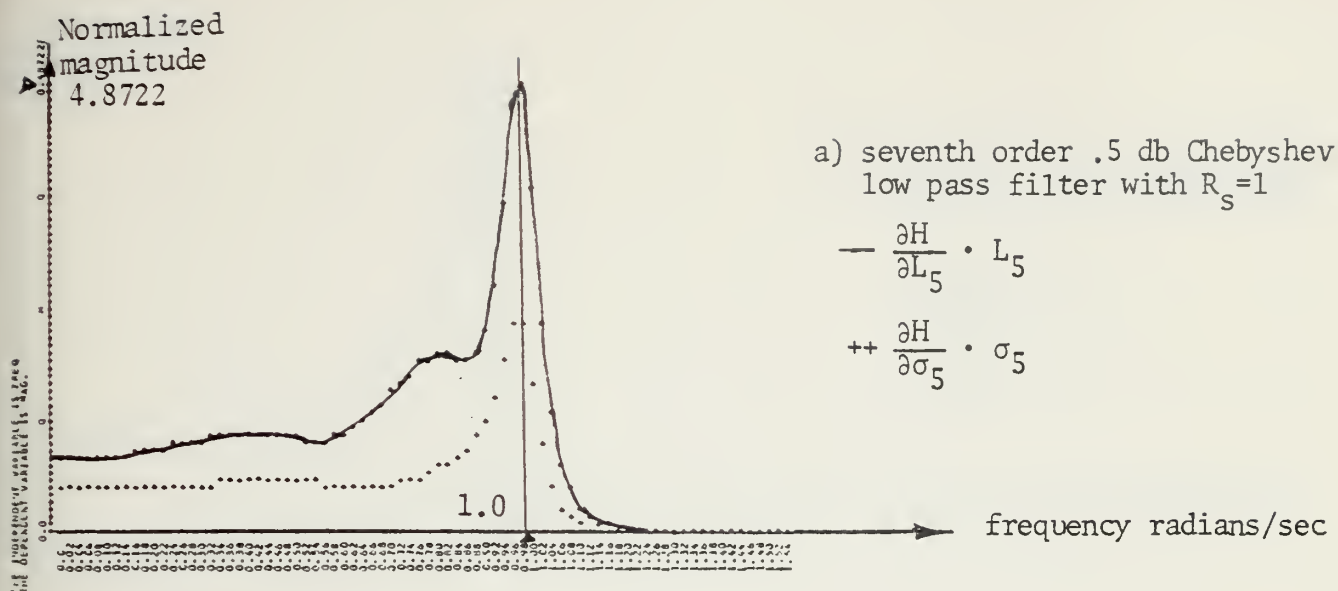


Fig. 6.10. The graphs of normalized sensitivity function,  $\frac{\partial H}{\partial L_5} \cdot L_5$  and  $\frac{\partial H}{\partial \sigma_5} \cdot \sigma_5$  of various simple wave digital filters.



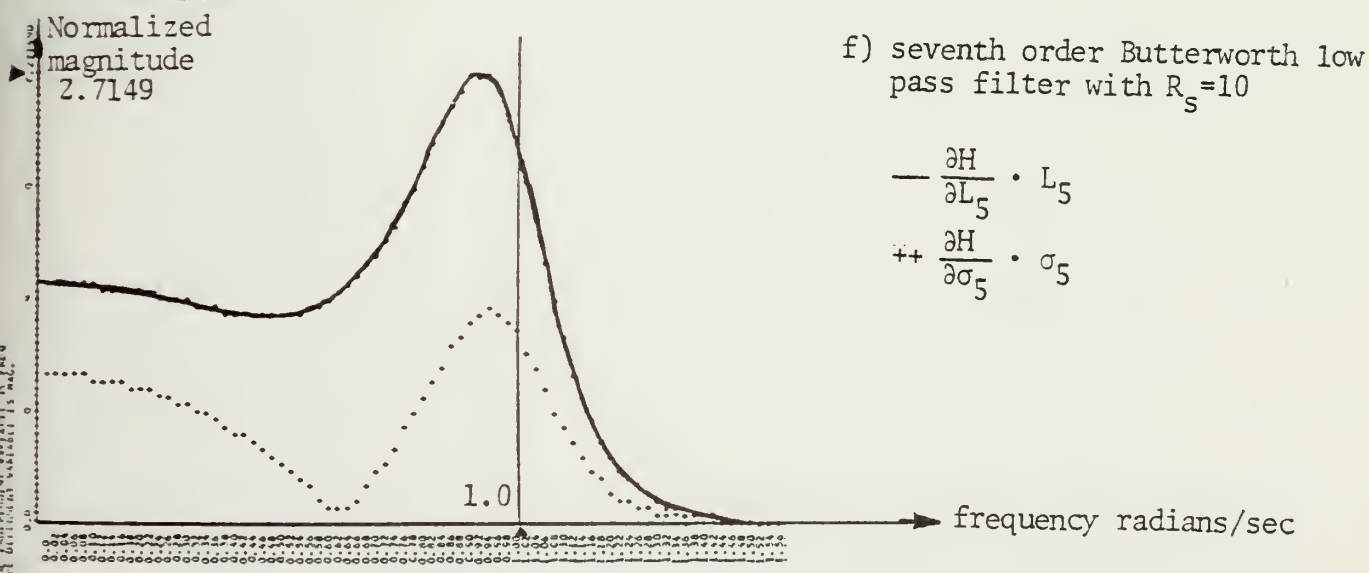
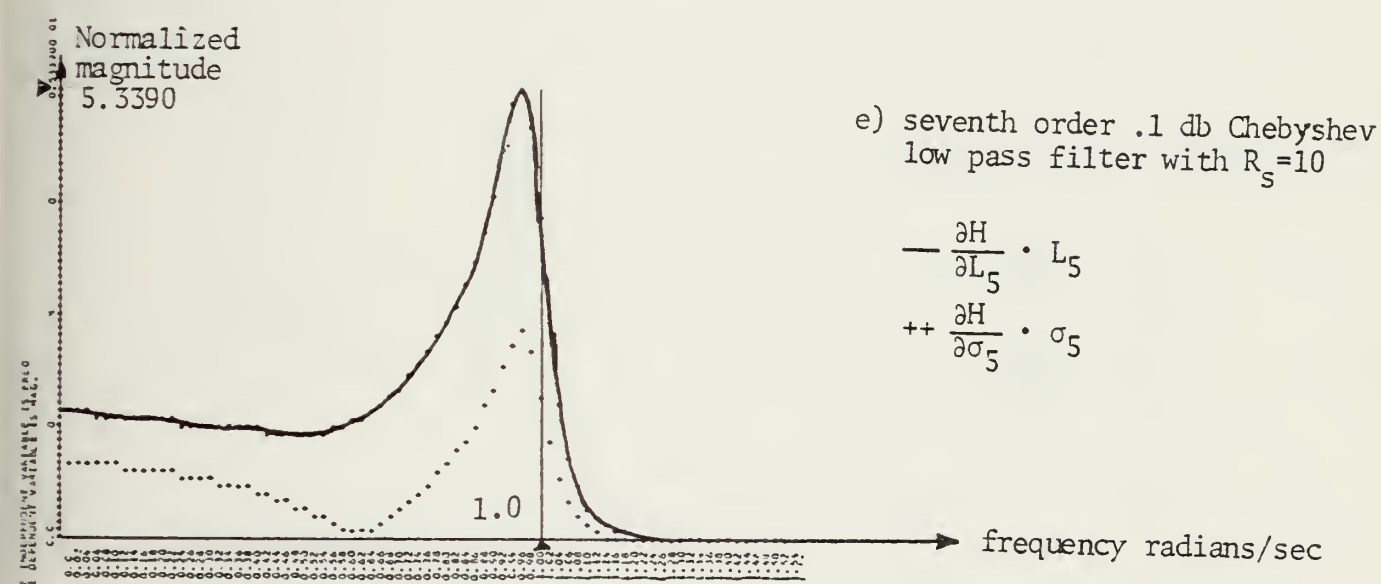
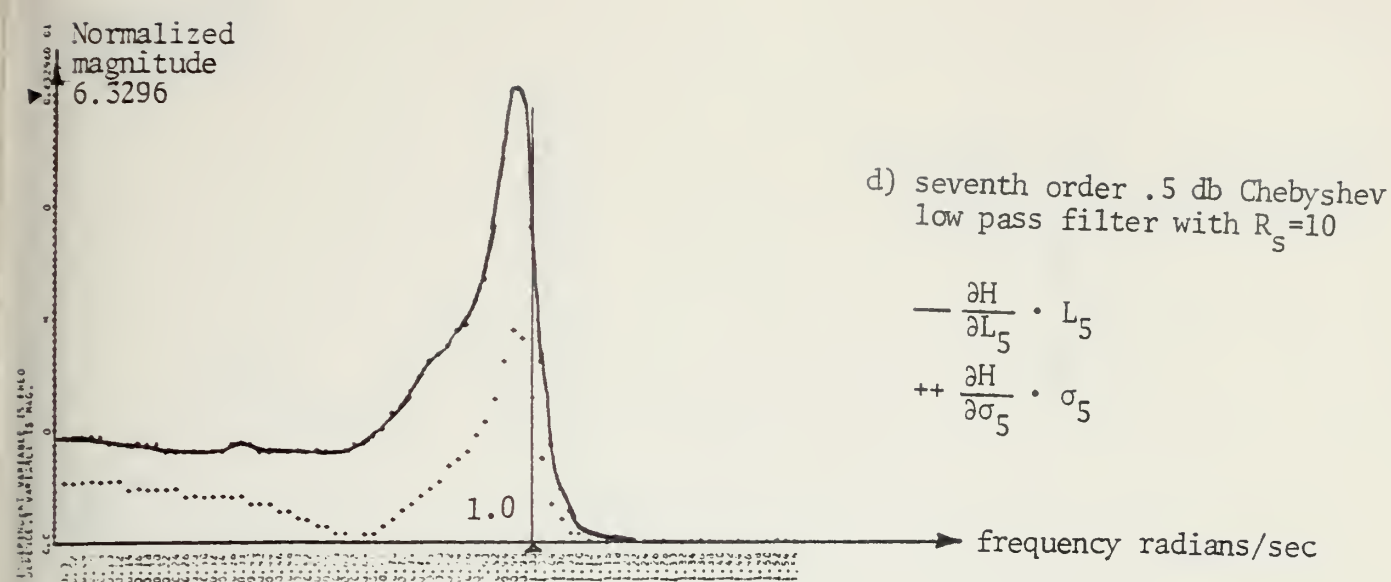


Fig. 6.10. Continued





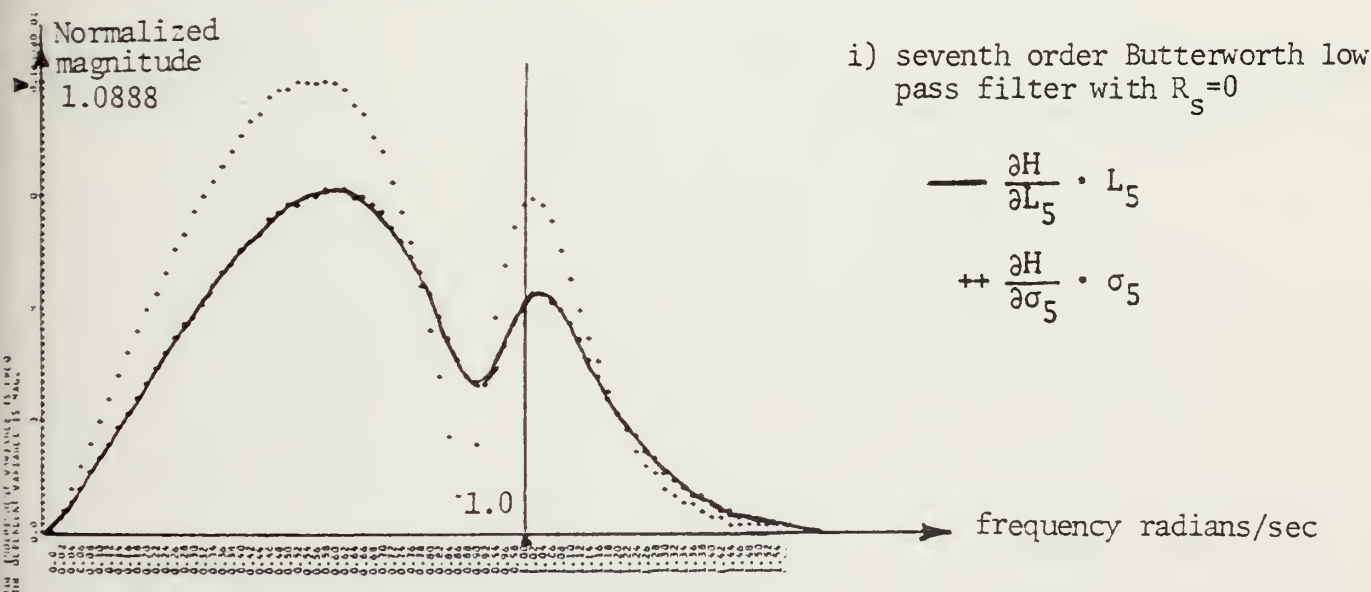
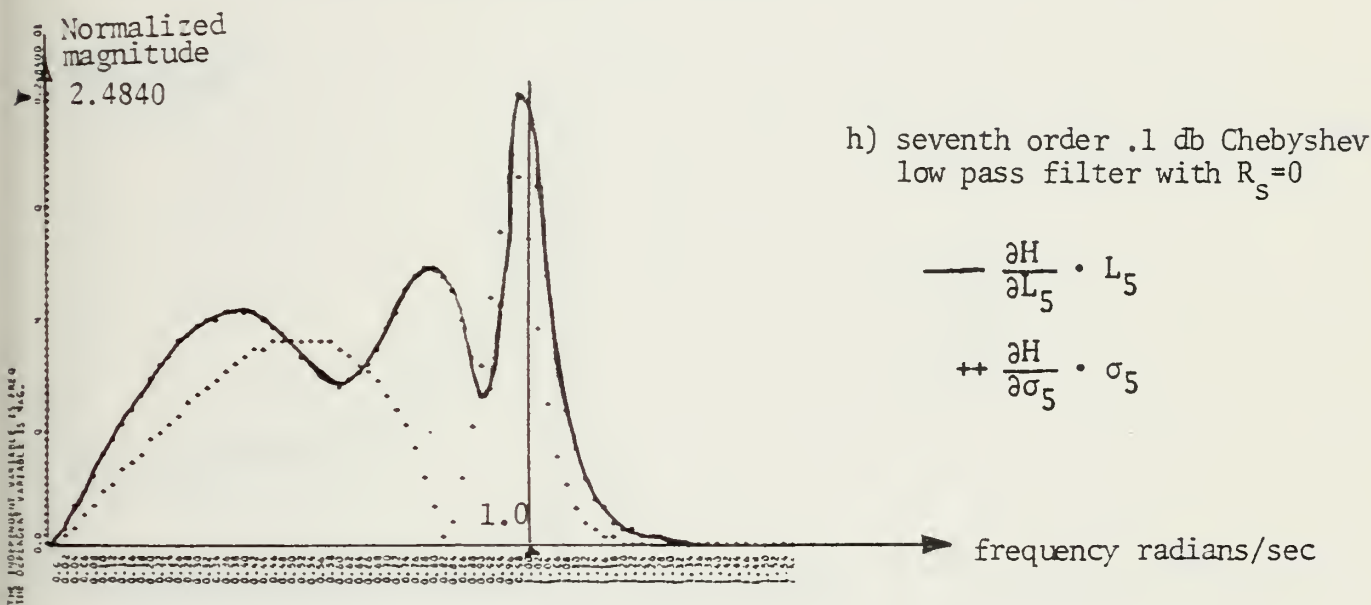
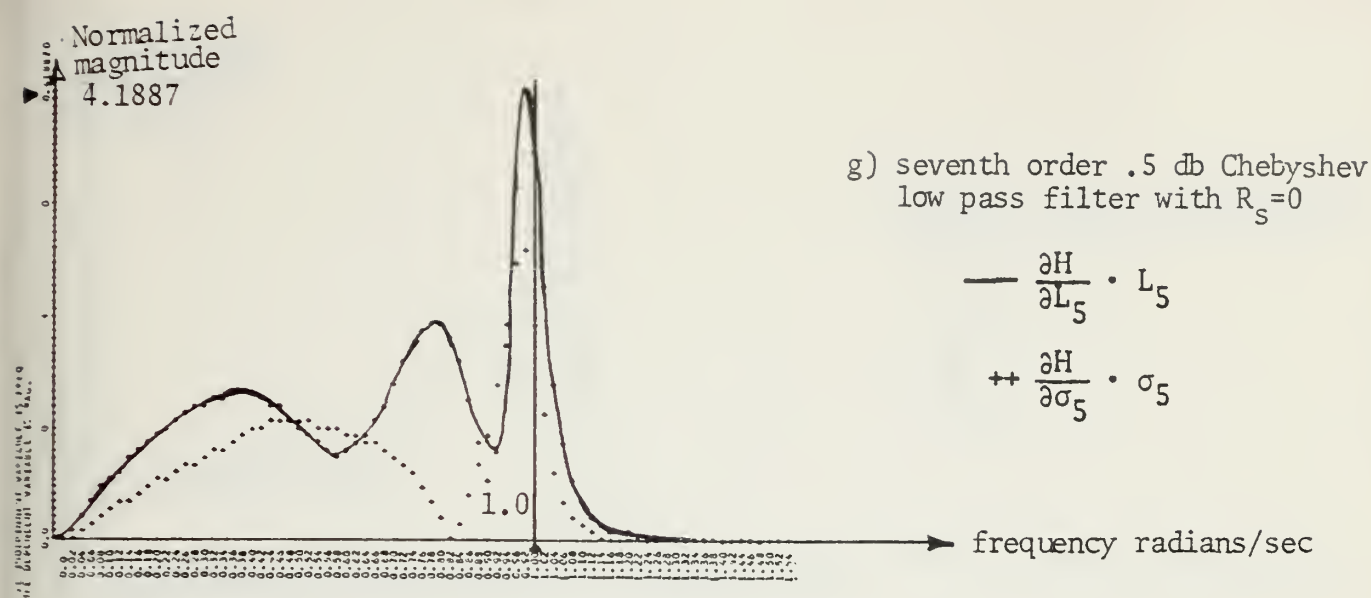


Fig. 6.10. Continued



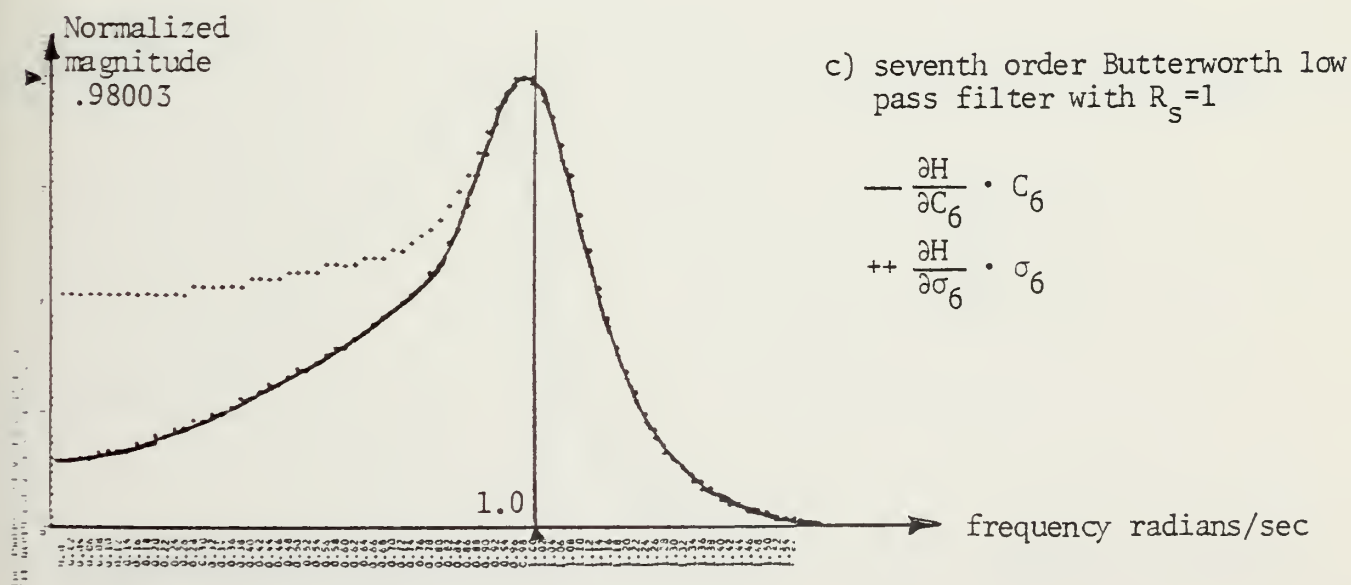
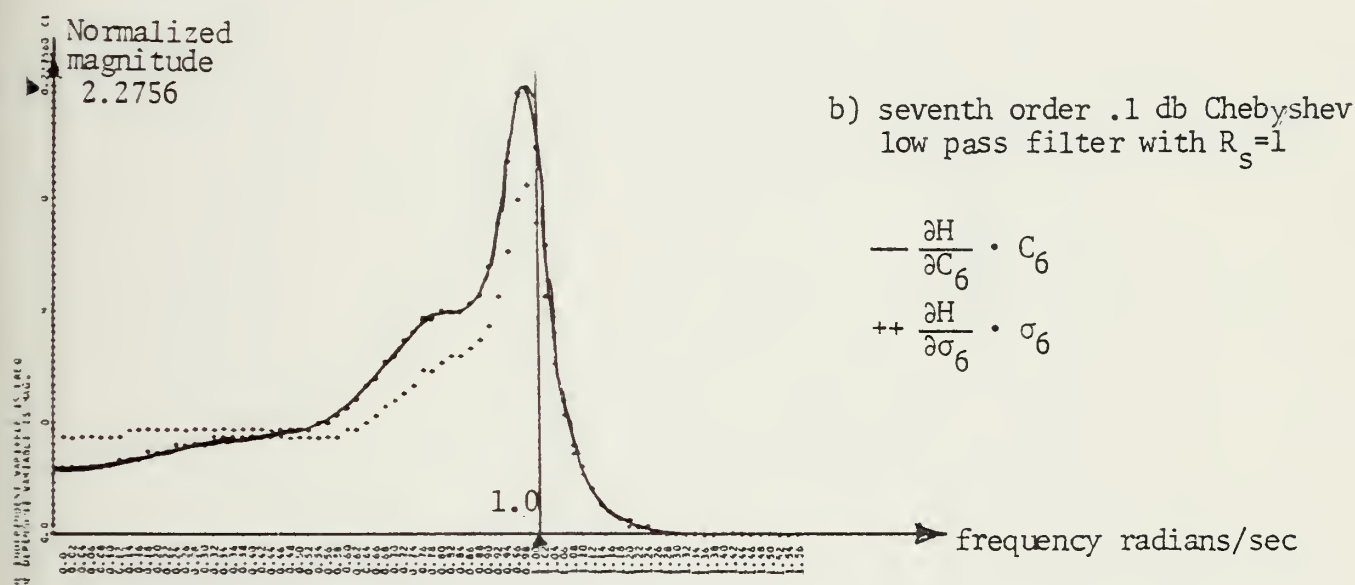
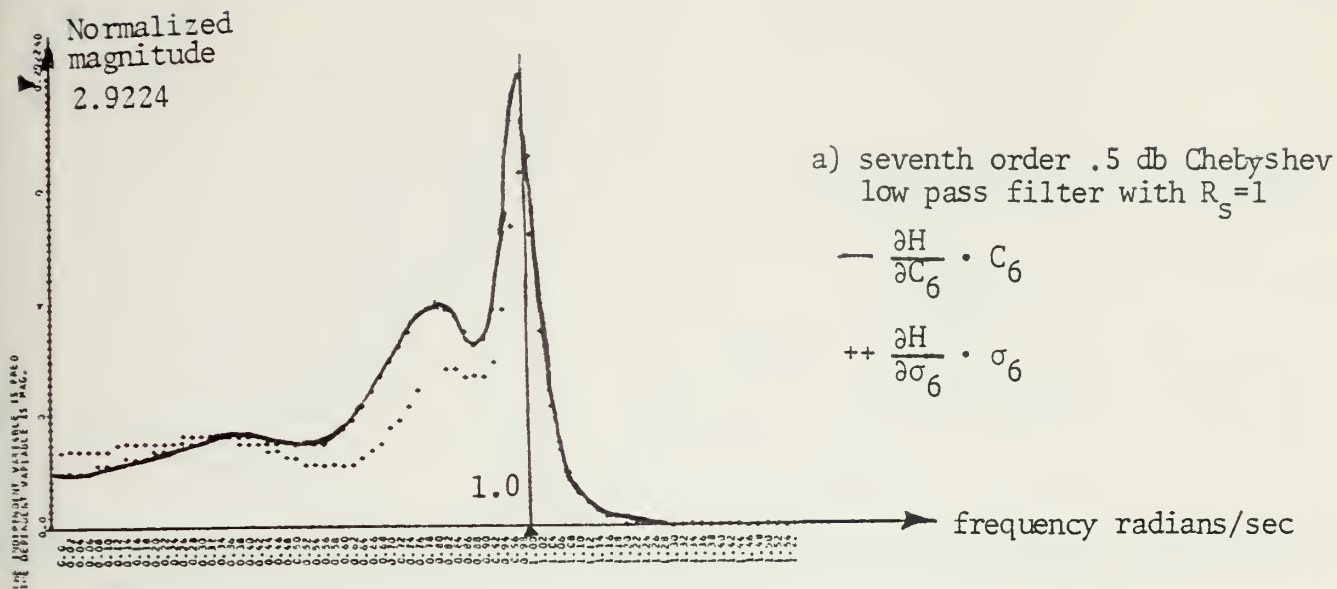


Fig. 6.11. The graphs of normalized sensitivity function,  $\frac{\partial H}{\partial C_6} \cdot C_6$  and  $\frac{\partial H}{\partial \sigma_6} \cdot \sigma_6$  of various simple wave digital filters.



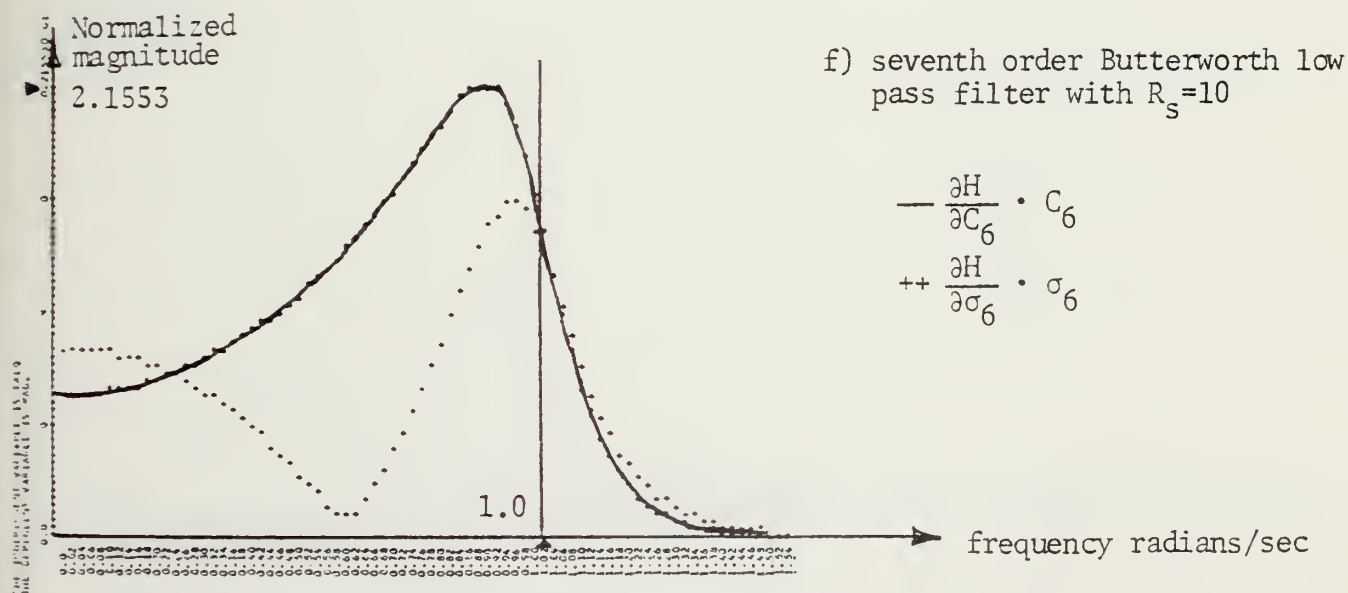
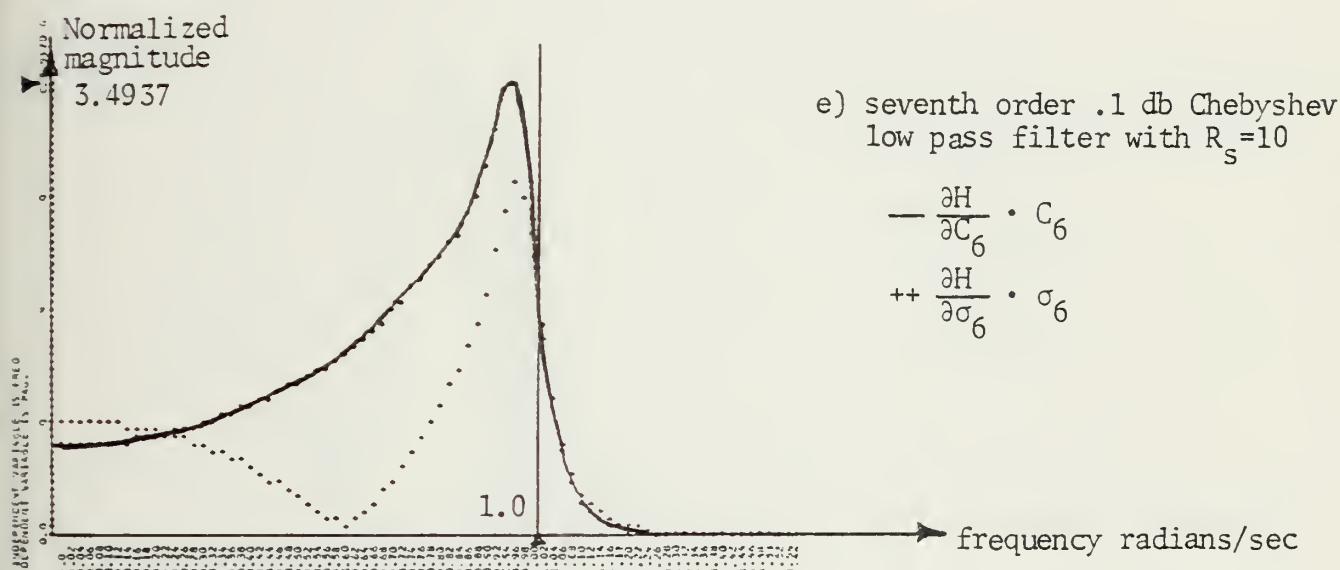
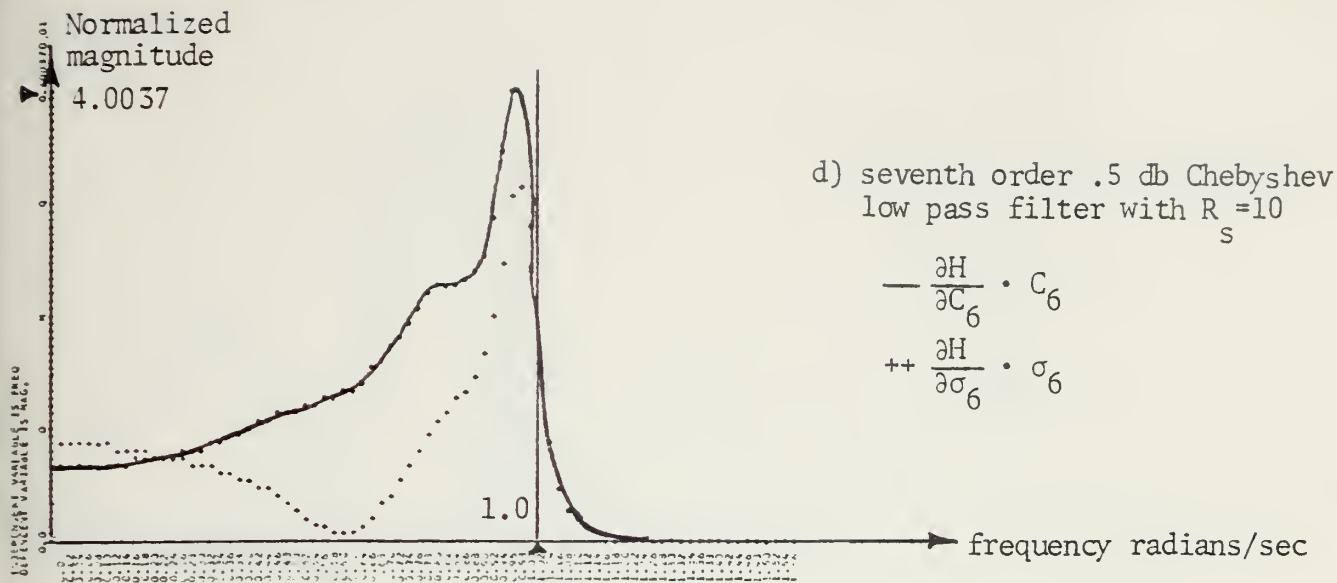


Fig. 6.11. Continued



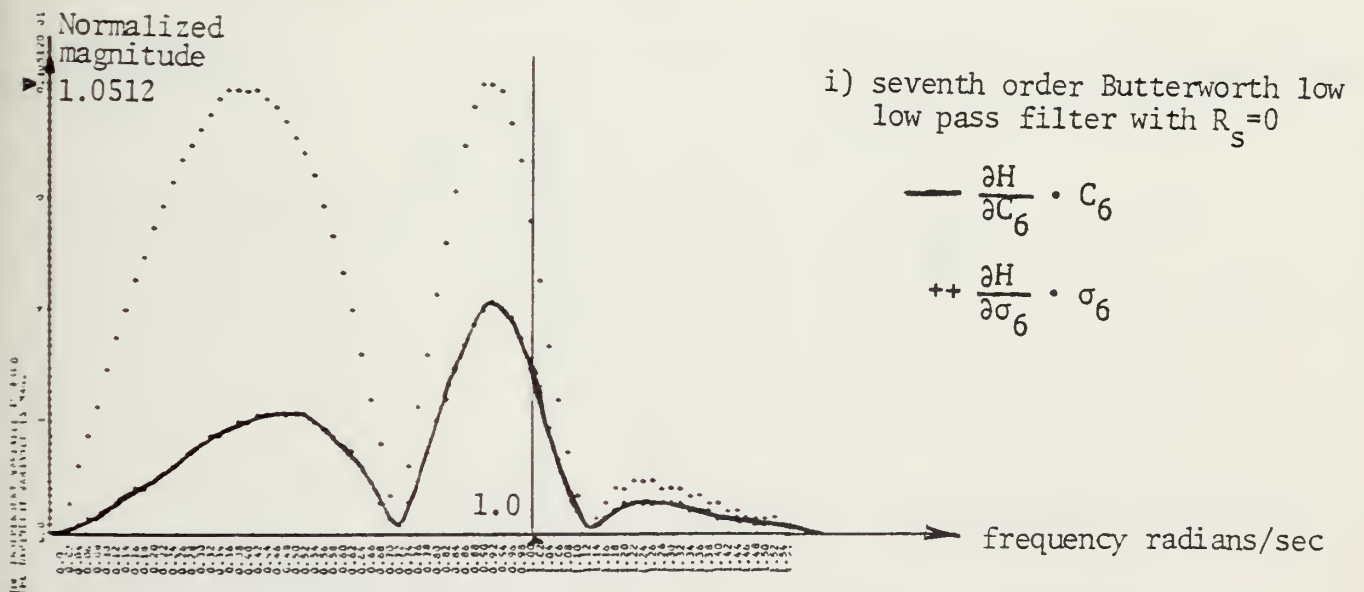
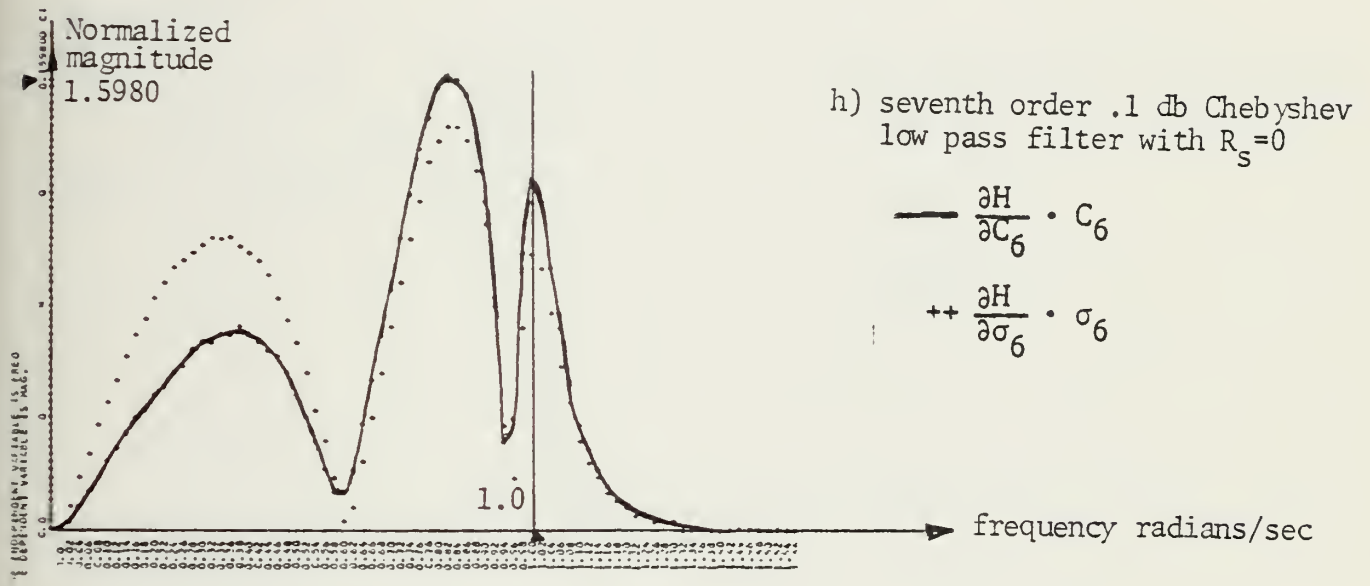
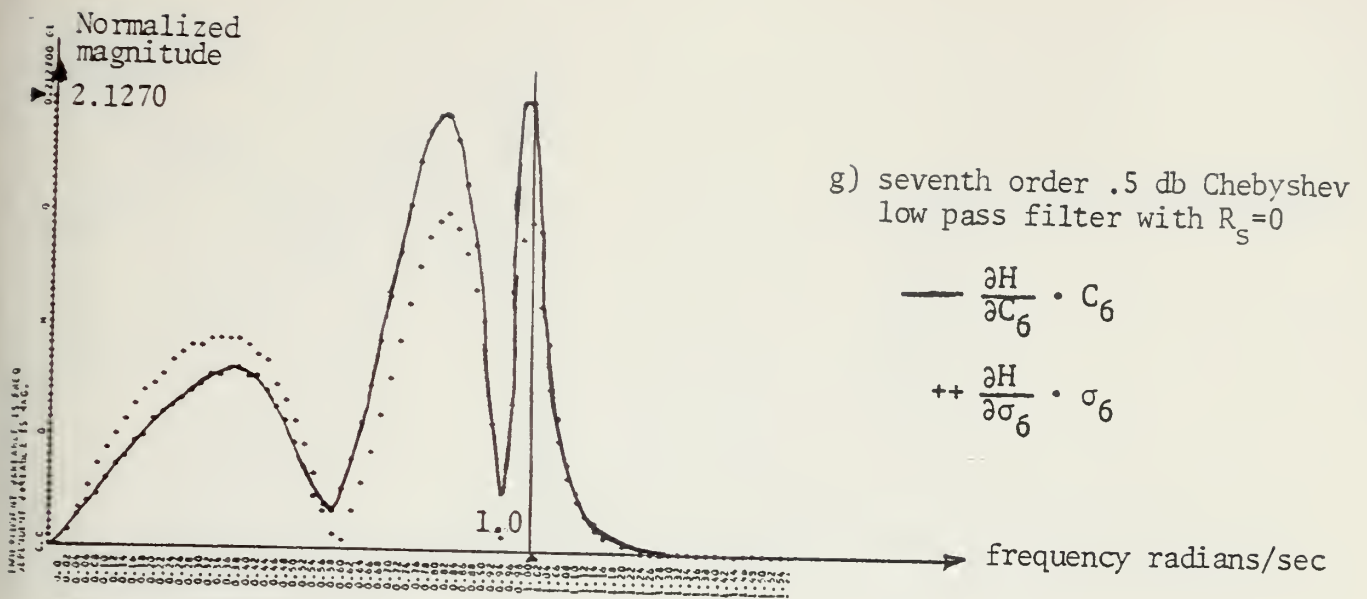
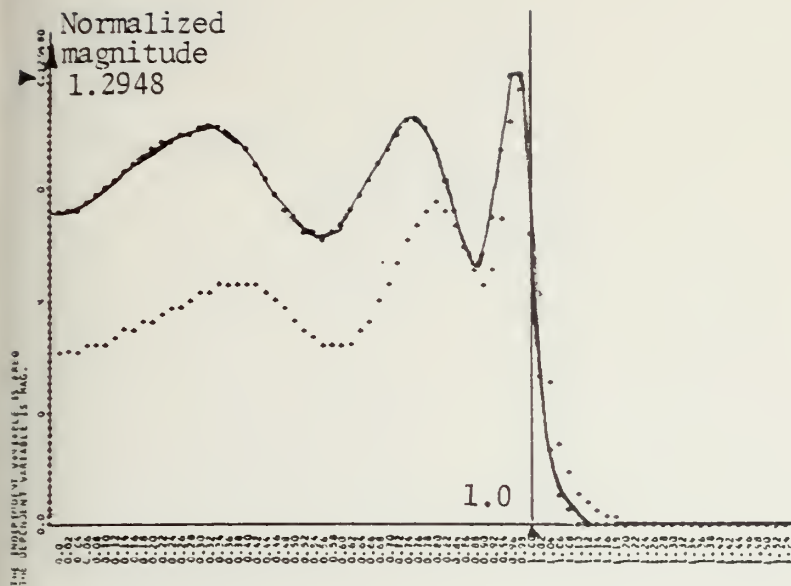


Fig. 6.11. Continued



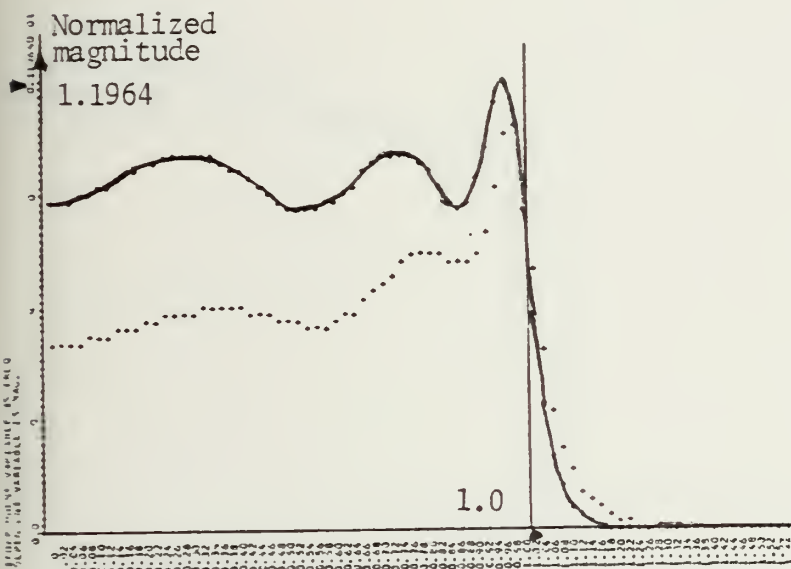




a) seventh order .5 db Chebyshev low pass filter with  $R_s=1$

$$- \frac{\partial H}{\partial L_7} \cdot L_7$$

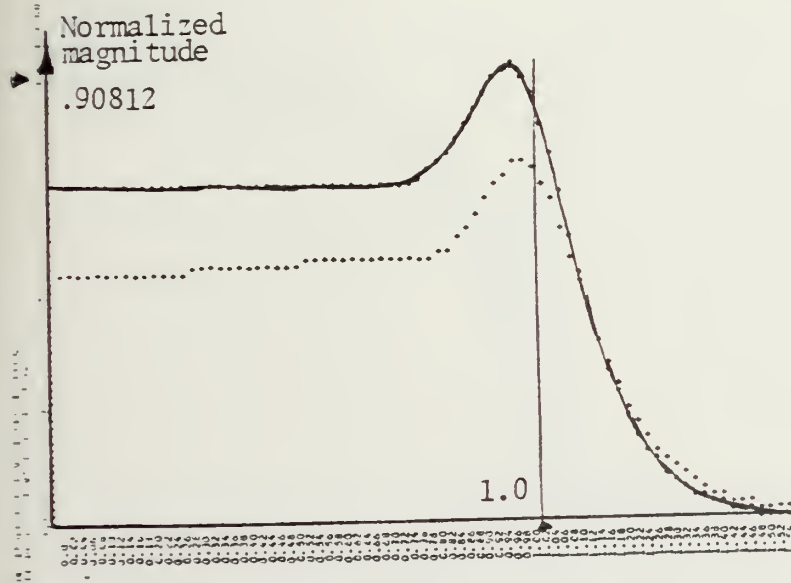
$$++ \frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$$



b) seventh order .1 db Chebyshev low pass filter with  $R_s=1$

$$- \frac{\partial H}{\partial L_7} \cdot L_7$$

$$++ \frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$$



c) seventh order Butterworth low pass filter with  $R_s=1$

$$- \frac{\partial H}{\partial L_7} \cdot L_7$$

$$++ \frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$$

Fig. 6.12. The graphs of normalized sensitivity function,  $\frac{\partial H}{\partial L_7} \cdot L_7$  and  $\frac{\partial H}{\partial \sigma_7} \cdot \sigma_7$  of various simple wave digital filters.



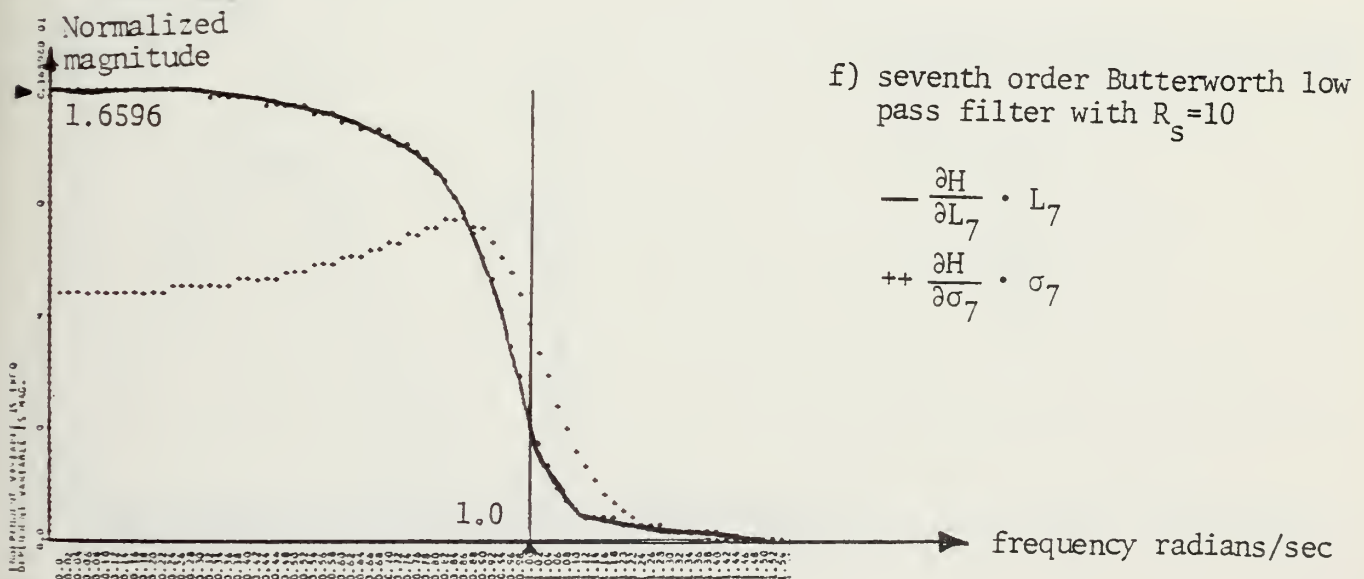
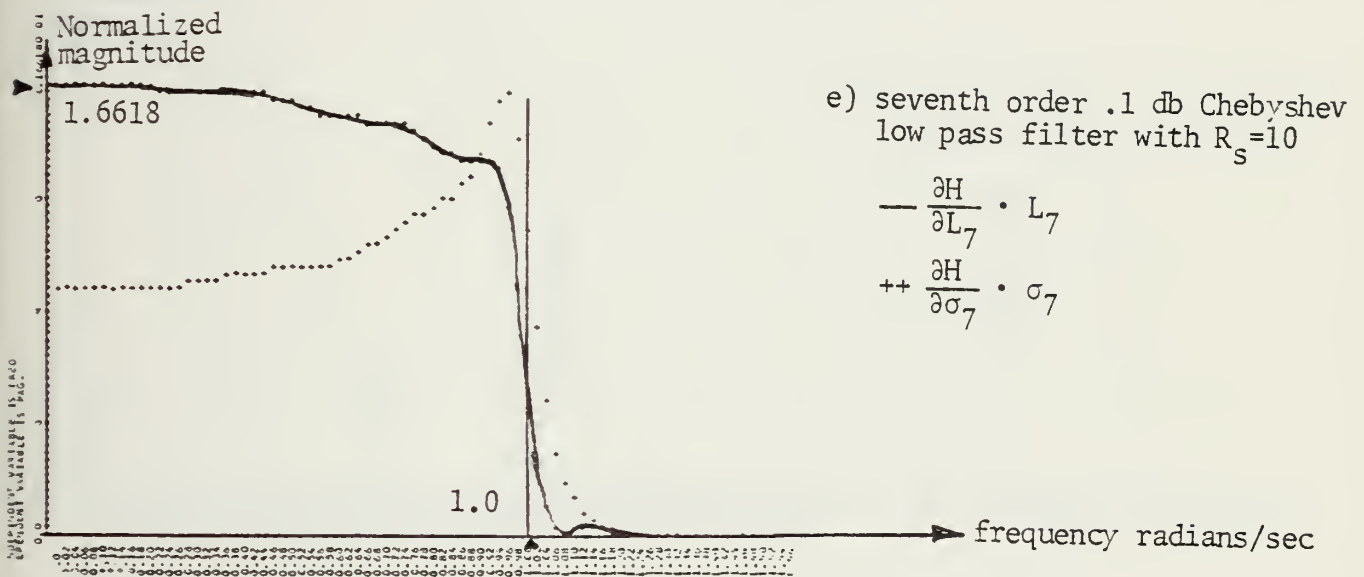
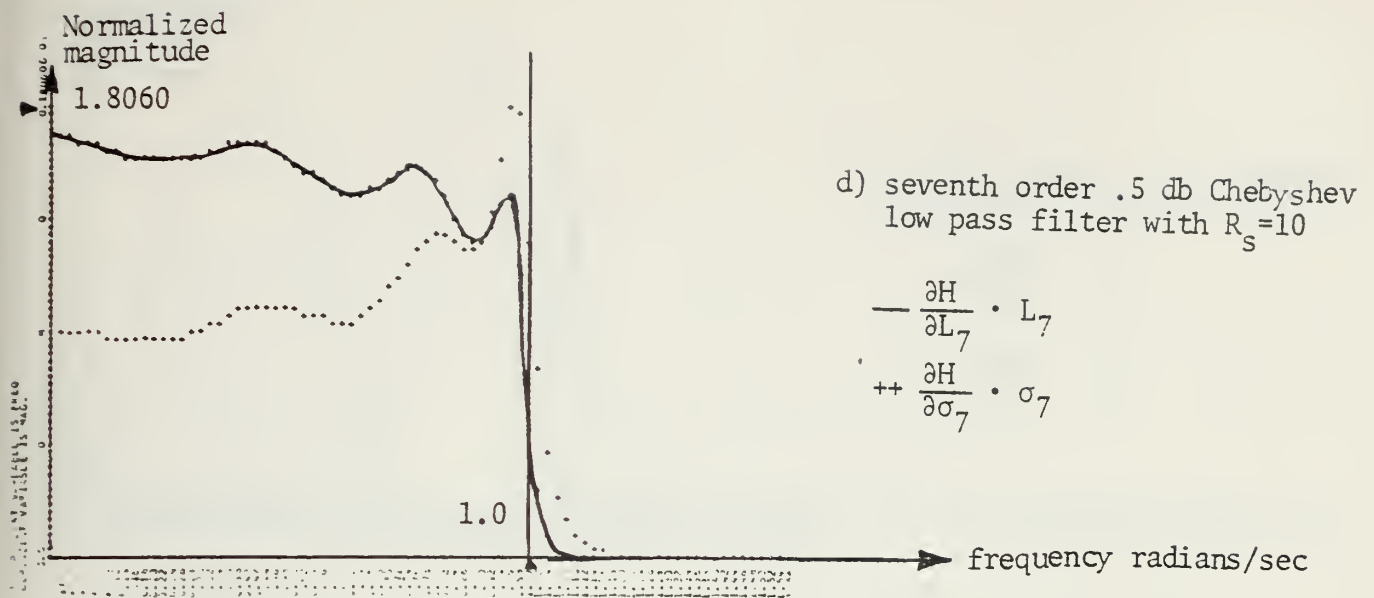


Fig. 6.12. Continued



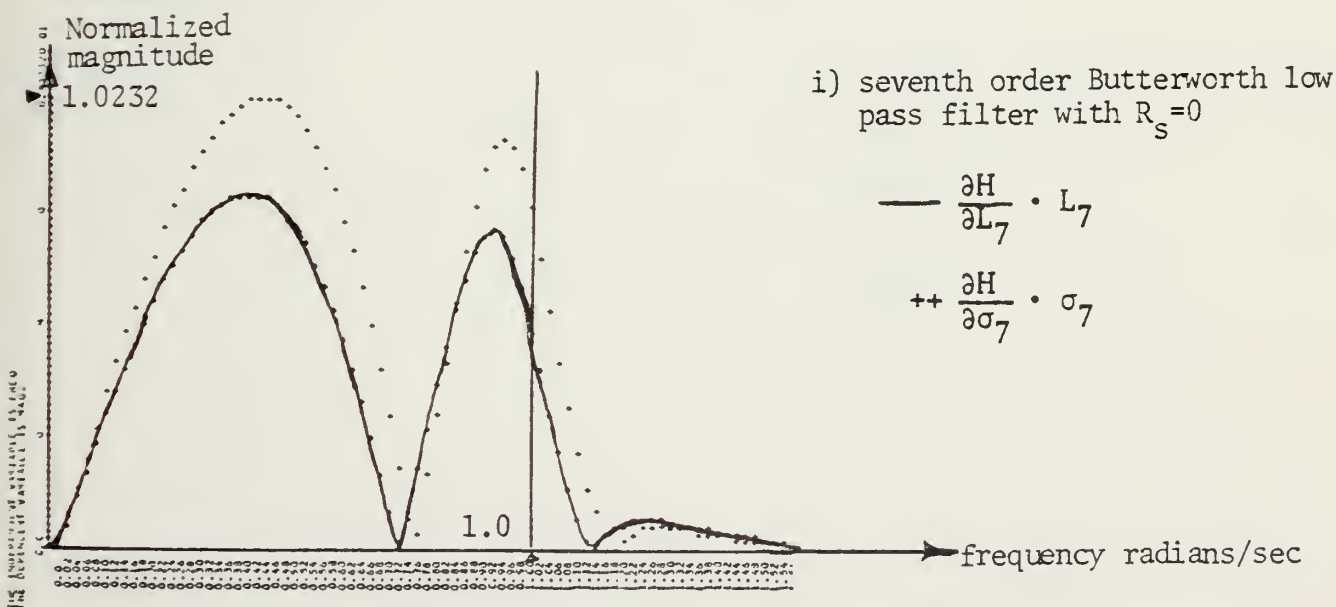
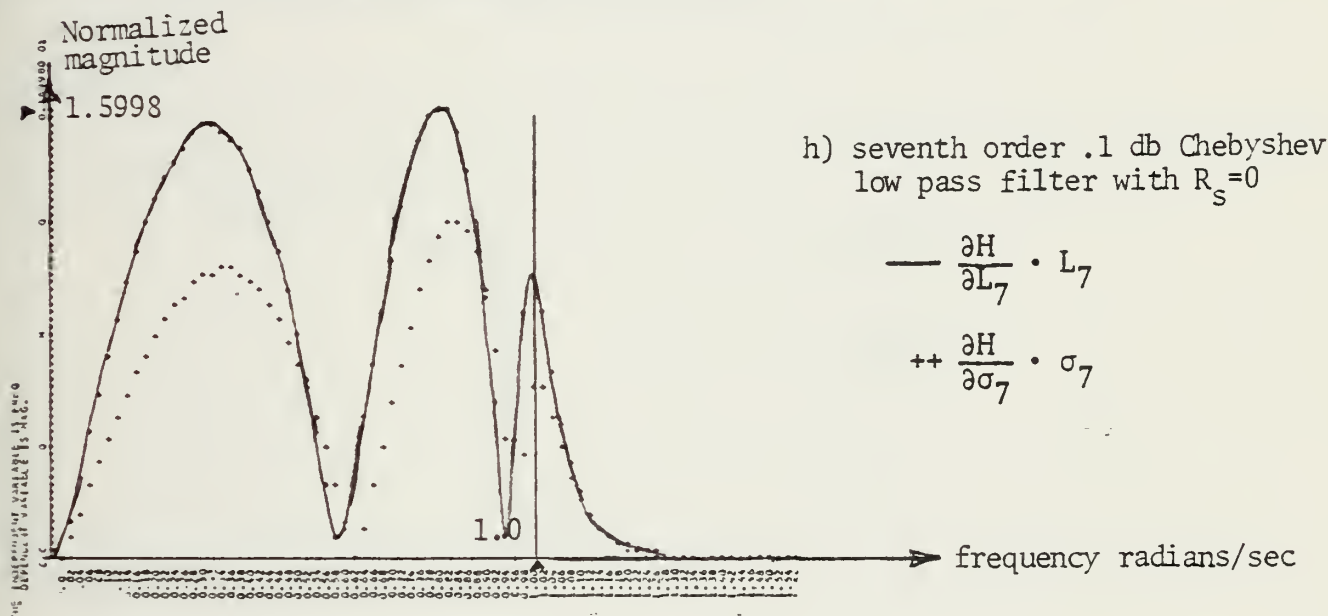
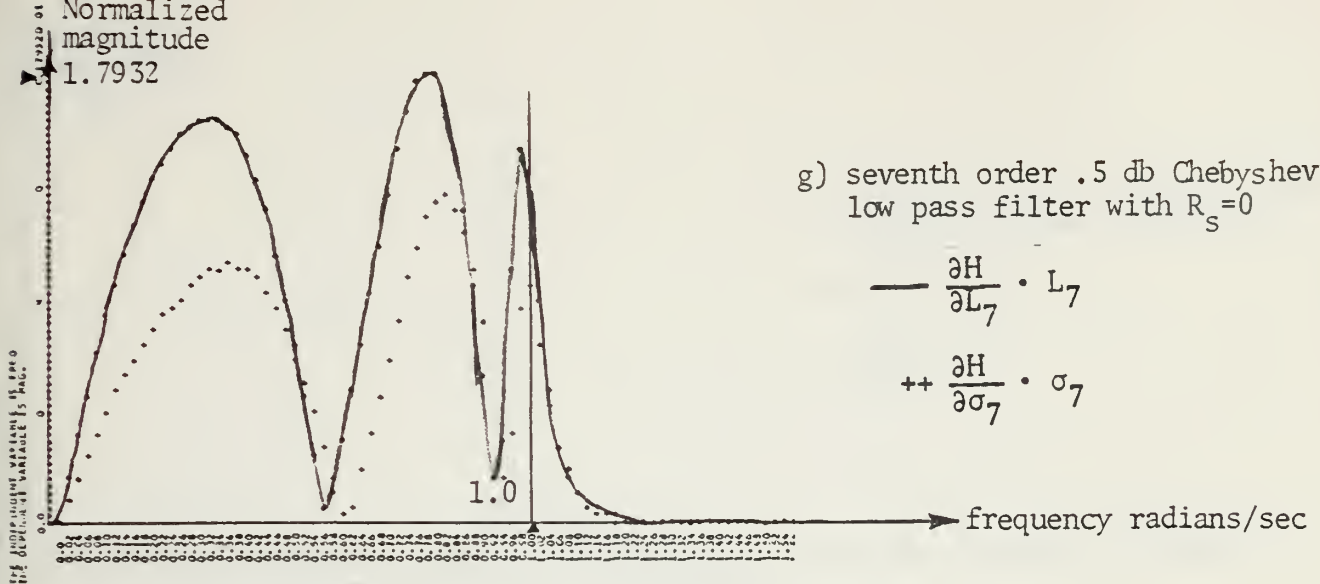


Fig. 6.12. Continued



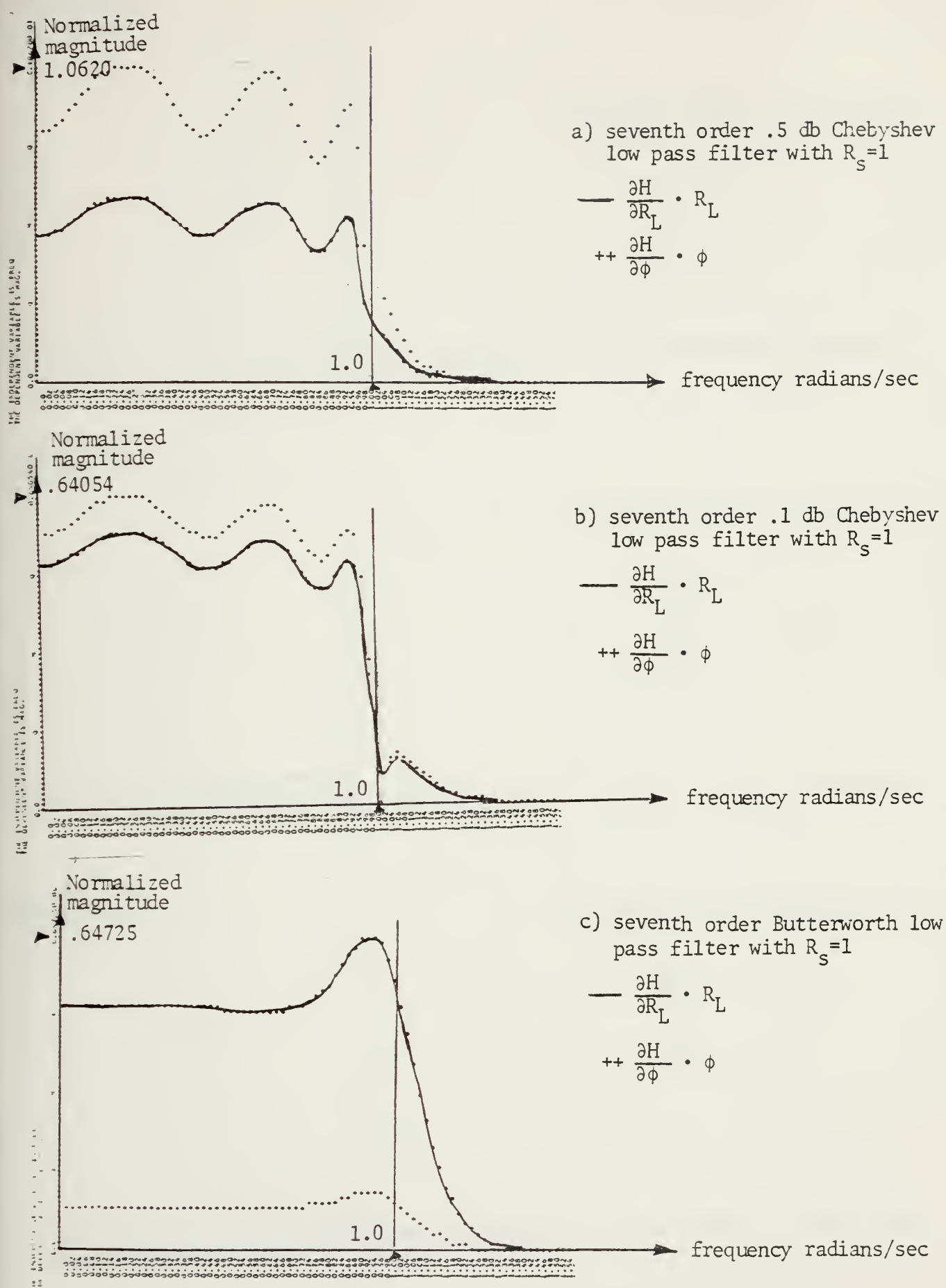
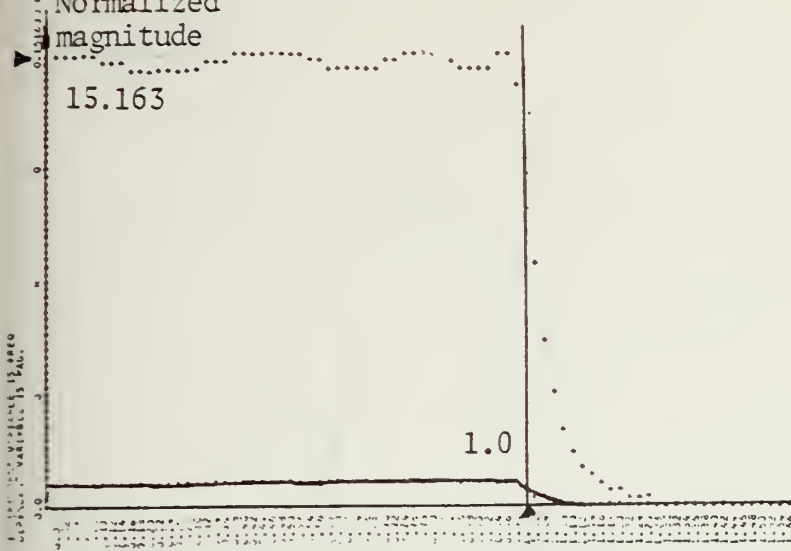


Fig. 6.13. The graphs of normalized sensitivity function, of various simple wave digital filters.

$$\frac{\partial H}{\partial R_L} \cdot R_L \text{ and } \frac{\partial H}{\partial \phi} \cdot \phi$$



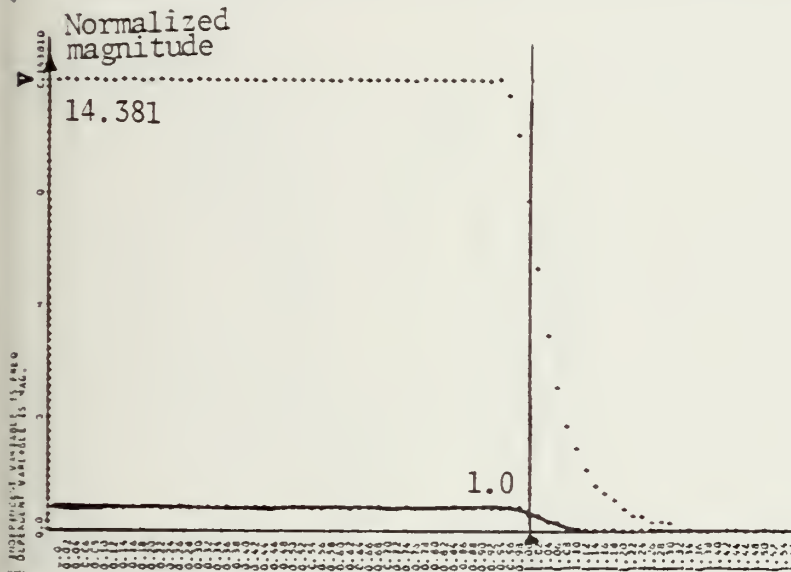




d) seventh order .5 db Chebyshev low pass filter with  $R_S=10$

$$- \frac{\partial H}{\partial R_L} \cdot R_L$$

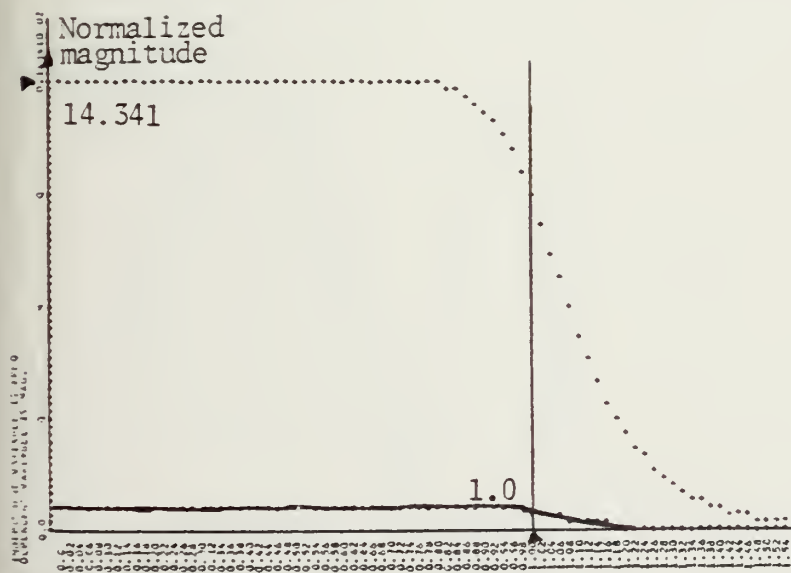
$$++ \frac{\partial H}{\partial \phi} \cdot \phi$$



e) seventh order .1 db Chebyshev low pass filter with  $R_S=10$

$$- \frac{\partial H}{\partial R_L} \cdot R_L$$

$$++ \frac{\partial H}{\partial \phi} \cdot \phi$$



f) seventh order Butterworth low pass filter with  $R_S=10$

$$- \frac{\partial H}{\partial R_L} \cdot R_L$$

$$++ \frac{\partial H}{\partial \phi} \cdot \phi$$

Fig. 6.13. Continued



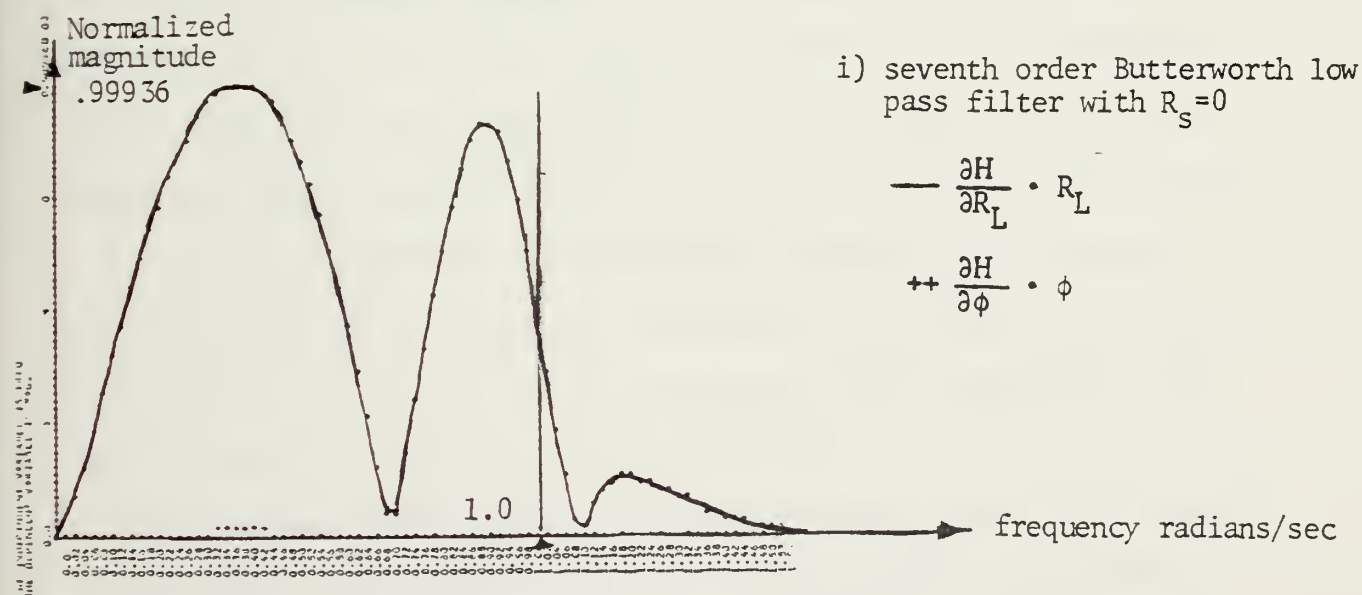
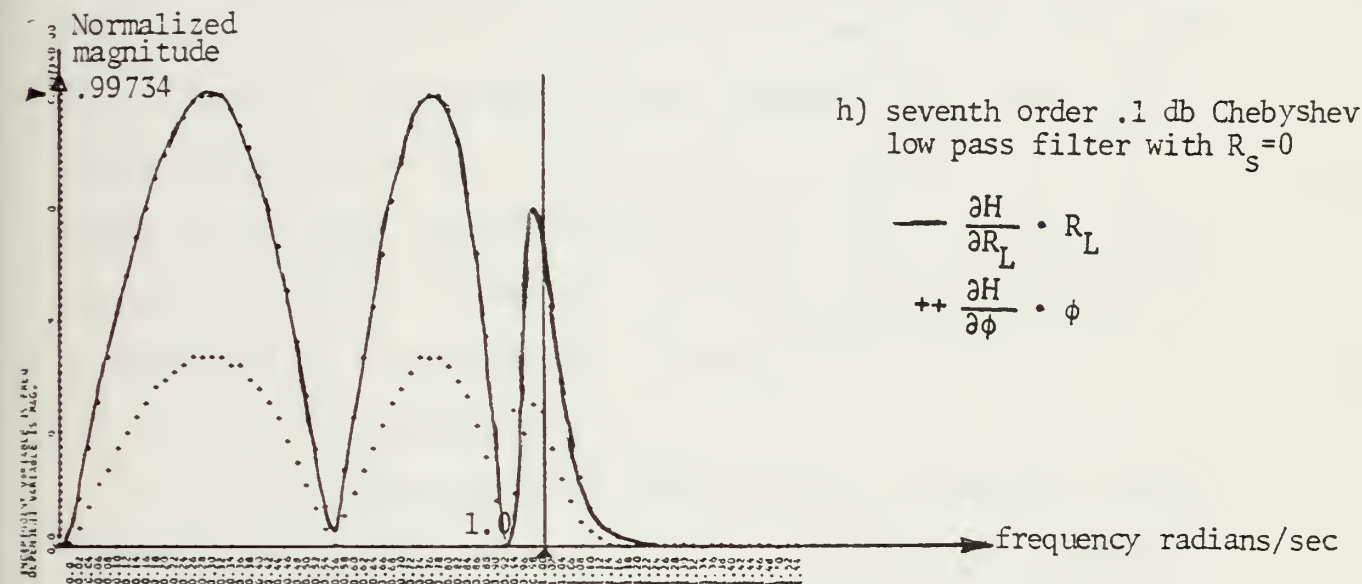
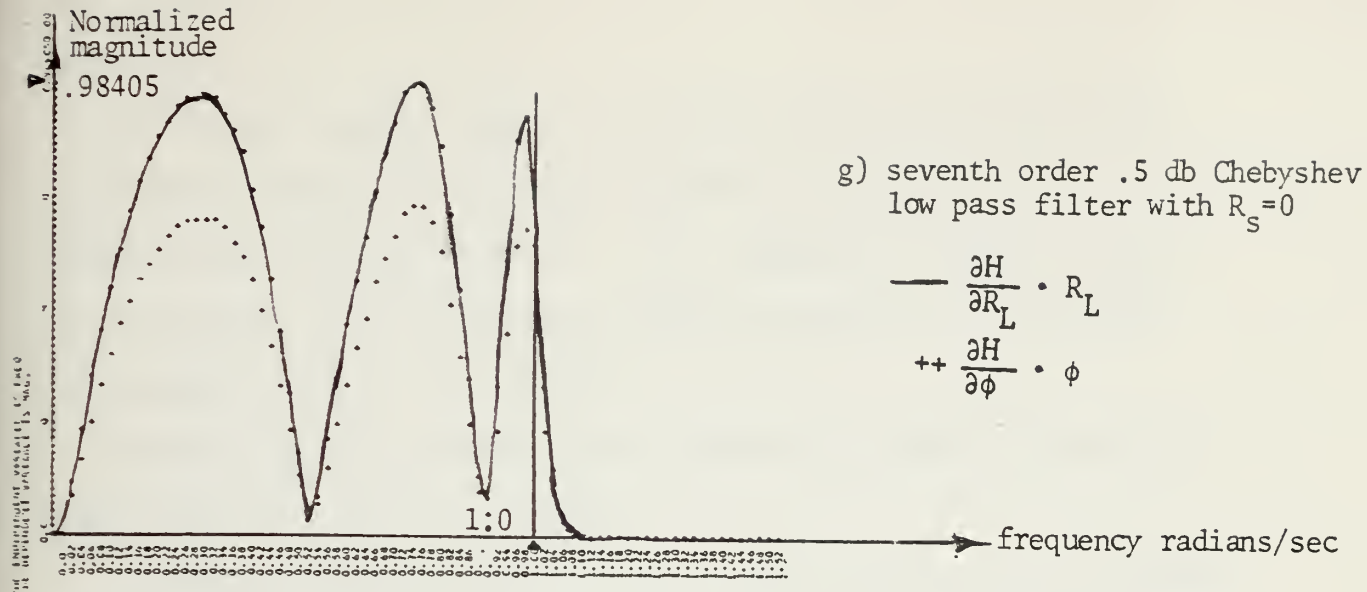


Fig. 6.13. Continued



in the frequency domain. Thus it is expected that the product of the sensitivity study of this chapter to conform to the bit truncation study of the Chapter V. It is important to emphasize that in this chapter only the internal sensitivity behavior of the simple wave digital filters is considered.

Analysis of the normalized partial sensitivity function curves of Figs. 6.5, 13 reveal the following.

1) Both sensitivity functions, i.e. sensitivity due to wave digital filter multiplier coefficients and sensitivity due to the original wave digital components peak at the critical frequency in all cases. This can be expected, since it is well known that the digital filters in general exhibit high sensitivities around the critical frequency of the filter. The critical frequency of the filters under investigation was 1 radian/sec and this point has been marked by a vertical line on all the sensitivity graphs in Figs. 6.5, 13.

2) In general the normalized sensitivity due to multiplier coefficients are smaller than sensitivity due to original filter components for the first sections. Gradually this difference gets smaller and for some few cases in the last section it reverses. This is reasonable since it can be expected that earlier multiplier coefficients will have lower sensitivity than the coefficients associated with the later sections.

3) At low frequencies, the normalized sensitivity of the wave digital filters due to multiplier coefficients are slightly larger than the sensitivity of the filter to the original filter components. This effect reverses in the mid and high frequencies.

4) The sensitivity function of the wave digital filter with respect to the source resistance increases with increasing source resistance, as evident from Figs. 6.5, 13.



5) The sensitivity with respect to terminating load resistance and reflection coefficient have exactly the same pattern as evident from equation (6.60). Apart from this fact it is important to note that both sensitivity functions tend to follow the frequency characteristic of the original filter. This can be seen from the graph 6.13 for the case  $R_s = 10$ . As  $R_s$  decreases the fluctuations in the sensitivity curve increases while following the same pattern. In fact for  $R_s=0$  these fluctuations increase to such an extent that at some frequencies the sensitivity becomes negligible. This phenomenon is also evident in the multiplier coefficients of the last few sections of the wave digital filter but to a lesser extent.

6) In general the sensitivity of the wave digital filter due to variations in multiplier coefficients is lower than the sensitivity of the wave digital filter with respect to the original component values.

7) Also it is worthwhile to note that the high sensitivities observed for high source resistances in wave digital filters algorithms with no delay free path in port two are in complete agreement with the results obtained in Chapter V.





## References

1. Anatol I. Zverev, Handbook of Filter Synthesis, 1967, published by John Wiley and Sons, Inc.



## VII. CONCLUSION

### A. INTRODUCTORY REMARKS

It is important to note that in this thesis, whenever appropriate, detailed conclusions and discussions of the results are made in the chapters concerned. Thus it would be repetitious to state these conclusions again. However for completeness, the highlights of important conclusions based on experimental results are summarized here. It is also important to note that in the experimental studies of this thesis, in order to achieve reliable and accurate results, a large number of filters of different types with different termination source resistances were studied, and the conclusions made on the basis of collective results.

### B. SUMMARY OF THE IMPORTANT RESULTS IN THE FREQUENCY DOMAIN BEHAVIOR OF THE WAVE DIGITAL FILTER

- 1 - Digital filters, derived from doubly terminated LC analogue filters using the bilinear transform, have the lowest sensitivity of frequency response for variation of the original  $L$ ,  $C$ ,  $R_s$  and  $R_L$  parameters of the algorithms tested.
- 2 - Wave digital filters, derived from doubly terminated LC analogue structures, if designed properly tend to achieve exactly the same low sensitivity as that of the above conventional digital filters.
- 3 - Design of conventional digital filters from LC structures is relatively a tedious job, while designing the wave digital filters from LC structures is relatively simple.



- 4 - Wave digital filters exhibit high sensitivity to termination resistance values. Thus in the design of wave digital filters, in order to achieve a desired performance, the delay free loop should be chosen at the low impedance termination of the filter.
- 5 - Due to the internal structure of the wave digital filter composed of sections with multiple LC resonant elements, these filters exhibit higher sensitivity than the same filter made up of sections with simple elements.
- 6 - The sensitivities of the internal sections of the wave digital filter, both with respect to filter multiplier coefficients and original filter RLC components, tend to peak sharply at the critical frequency of the wave digital filter.
- 7 - The sensitivities of the internal sections of the wave digital filter increases towards the load end.
- 8 - For the seventh order low pass wave digital filters studied, the slope of the rms error due to quantization in the number of bits of the multiplier coefficients versus the number of bits is approximately 3 db per bit. Also, interestingly, the slope of the rms error of the conventional seventh order digital filter is also approximately 3 db per bit.

#### C. SUMMARY OF NEW THEORETICAL EXTENSIONS TO WAVE DIGITAL FILTER THEORY

- 1 - Earlier researchers [1] and [2] have stated as a typical example that for the port two resistance of the wave digital section designed for series L with no delay free path in port two, one should use  $R_2 = R_1 + L$ . This formula does not take into account the effect of sampling time. In the theory developed in this



thesis, following a derivation parallel to the one given in [1] and [2], the port two resistance for the said section is derived as  $R_2 = R_1 + \frac{2L}{T}$ . This result is more general than the previous one, for which an inherent fixed sampling time of one second must always be assumed.

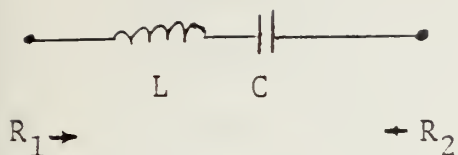
- 2 - In the theory developed in this thesis for the design of wave digital subsections, only one algorithm is derived for a multiple LC resonant section. This result is applicable for both series or shunt elements. This new approach makes it possible to design several alternate wave digital filter algorithms, for a given analogue LC filter structure.

#### D. SUGGESTED FUTURE RESEARCH

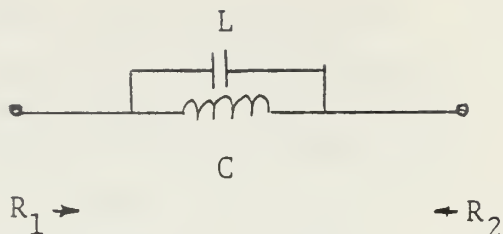
- 1 - In the analysis of this thesis we have established that the simple wave digital filter, due to its inherent ladder-like structure, exhibits lower sensitivities to multiplier truncation than the complex cascaded section wave digital filter. Thus the need for the design of simple wave digital filters, even for band pass or band stop applications, arises. In the literature all the present algorithms for band pass or band stop wave digital filters are of the complex cascaded multiple LC element type. However it is possible to design simple wave digital filter, even for band pass or band stop applications using simple sections, so as to reduce the sensitivity of the overall structure. The technique basically is as developed in Chapter IV. The suggested outline is as follows. The analogue band pass or band stop LC filter derived from the low pass analogue LC filter would have sections of the type shown in Fig. 7.1. The design



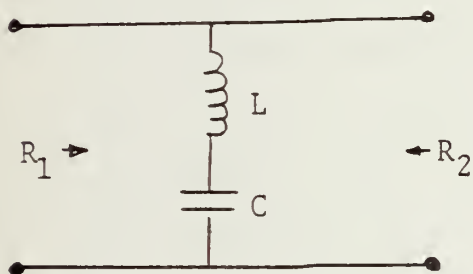




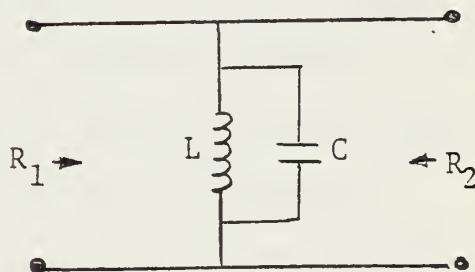
a) two port network with two L and C elements in the total series configuration



b) two port network with parallel L and C in the total series configuration



c) two port network with series L and C in the total parallel configuration



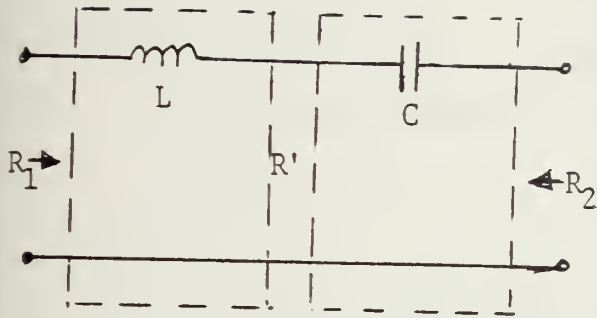
d) two port network with parallel L and C in the total parallel configuration

Fig. 7.1. Total combination of L and C in a two port network.

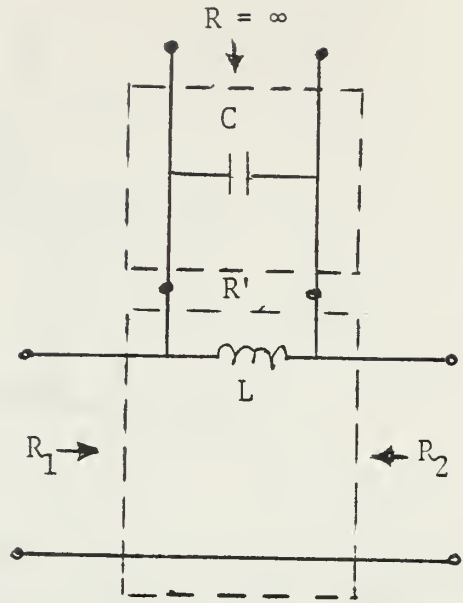


of simple wave digital filter for Figs. 7.1a and 7.1d is no problem and they can be separated into two port simple sections as shown in Figs. 7.2a and 7.2d. Thus we have to discuss only the cases of Figs. 7.1b and 7.1c. To design a simple wave digital algorithm for Fig. 7.1b we must employ a three port wave flow network. A general three port wave flow network is shown in Fig. 7.3. Note that in this figure there is a feedback path from all inputs to all outputs. Fig. 7.2b reveals that in order to design two simple wave digital sections we have to employ a three port signal flow graph. Either L or C of Fig. 7.2b can be adapted into a three port structure. Consider element L as the three port element. It can be shown that in order to have a causal network, two of the three ports must have no delay-free feedback path. We can make port-two and port-three of Fig. 7.3 with no delay free feedback. In doing so we can find  $R_2$  in terms of  $R_1$ ,  $R_3$  and L. Also we can find  $R_3$  in terms of  $R_1$ ,  $R_2$  and L. Thus we have two equations and two unknowns and we can solve for  $R_2$  and  $R_3$  in terms of  $R_1$  and L. Now we can cascade a two port network designed for the component C into port three, and also cascade the succeeding sections into port two. It must be emphasized that the two port network designed for element C has no termination resistance on its second port; thus it is open circuited and its reflection coefficient in port two will be equal to 1. To design a simple section wave digital filter for Fig. 7.1c, we have to use a somewhat different approach. With the concept of delay free feedback in mind and with reference to Fig. 7.4, we can design a two port network

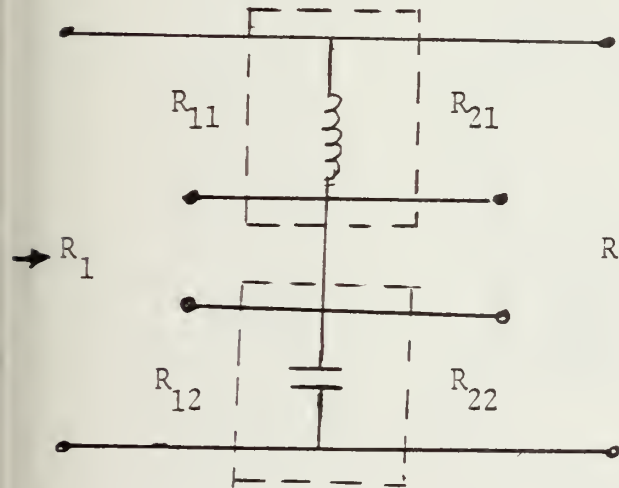




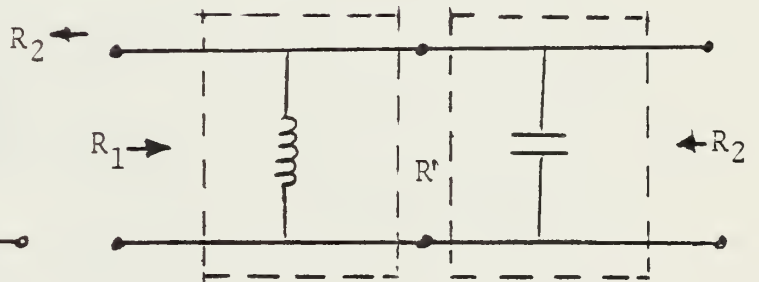
a) two cascaded two ports



b) three port element connected to the two port element



c) two series two ports in shunt configuration



d) two cascaded two ports

Fig. 7.2. Separation of the complex two port networks shown in Fig. 7.1 into simple two or three port networks.



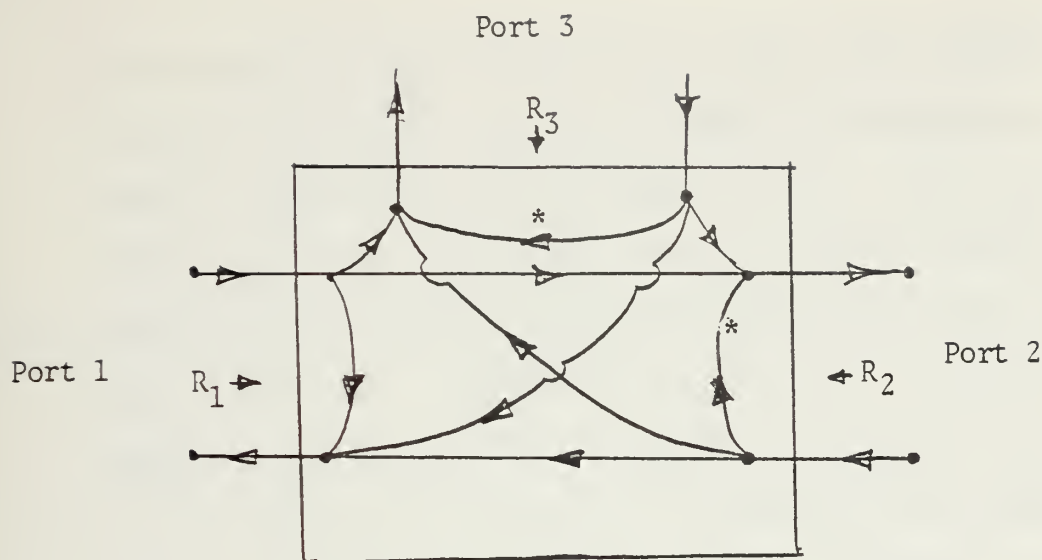


Fig. 7.3. A three port signal flow graph, with all possible combinations of delay free inputs and outputs. Note that the delay free path in port two and port three are marked with \*.

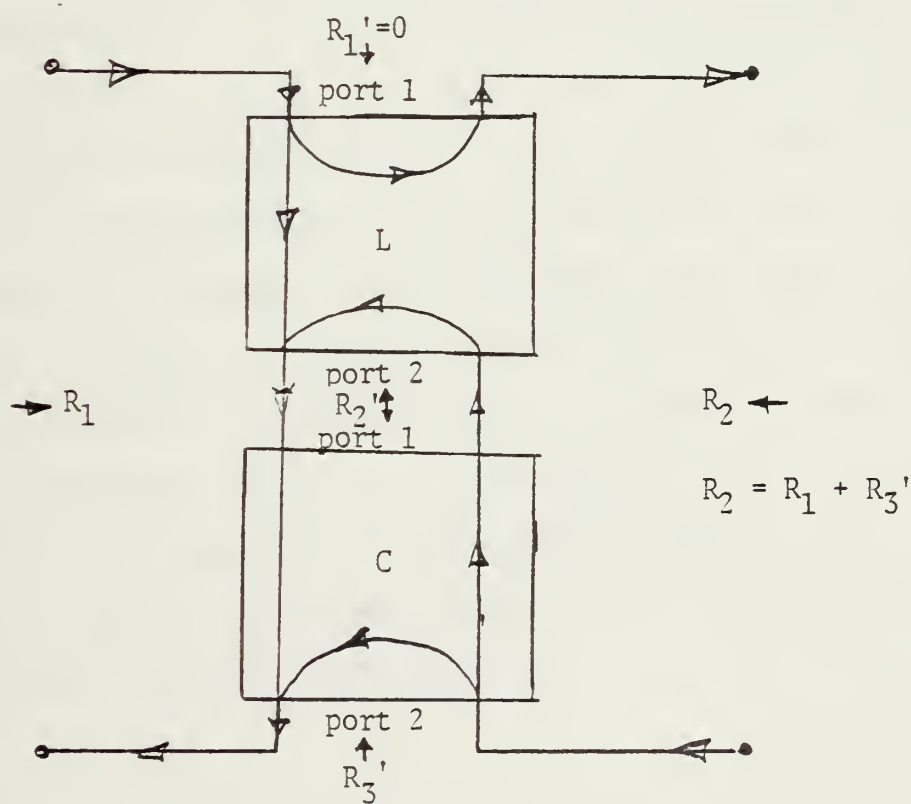


Fig. 7.4. Simple section wave digital filter signal flow graph of the complex series, shunt network of Fig. 7.1c.





for the element L such that there is no delay free feedback from port two (input) to port one (output) for subelement L. From this constraint, we can find  $R_2'$ , with  $R_1'$  arbitrarily made equal to zero. The signal flow graph for the element C then is a normal two port signal flow graph with no delay free path in port one. It is fairly obvious that port two of the composite section will have a termination resistance of  $R_2 = R_1 + R_3'$  where  $R_3'$  is the port two resistance of the element C.

- 2 - In the analysis of the internal structure of Chapter VI, we found the sensitivities of the wave digital filter with respect to the original component values, i.e. L's, C's,  $R_s$ , and  $R_L$ . It is interesting, for a given filter, to differentiate the conventional digital filter transfer function with respect to components L, C,  $R_s$ , and  $R_L$  and compare these sensitivities to those of the wave digital filter obtained in Chapter VI.
- 3 - The sensitivity functions of the wave digital filter with respect to filter multiplier coefficients can be used to implement an adaptive wave digital filter, by feeding back a weighted percentage of the sensitivities into the input as shown schematically in Fig. 7.5. The adaptivity can be done using any of the known optimization methods available in the literature, namely gradient optimization techniques, Fletcher-Powell optimization techniques, etc.



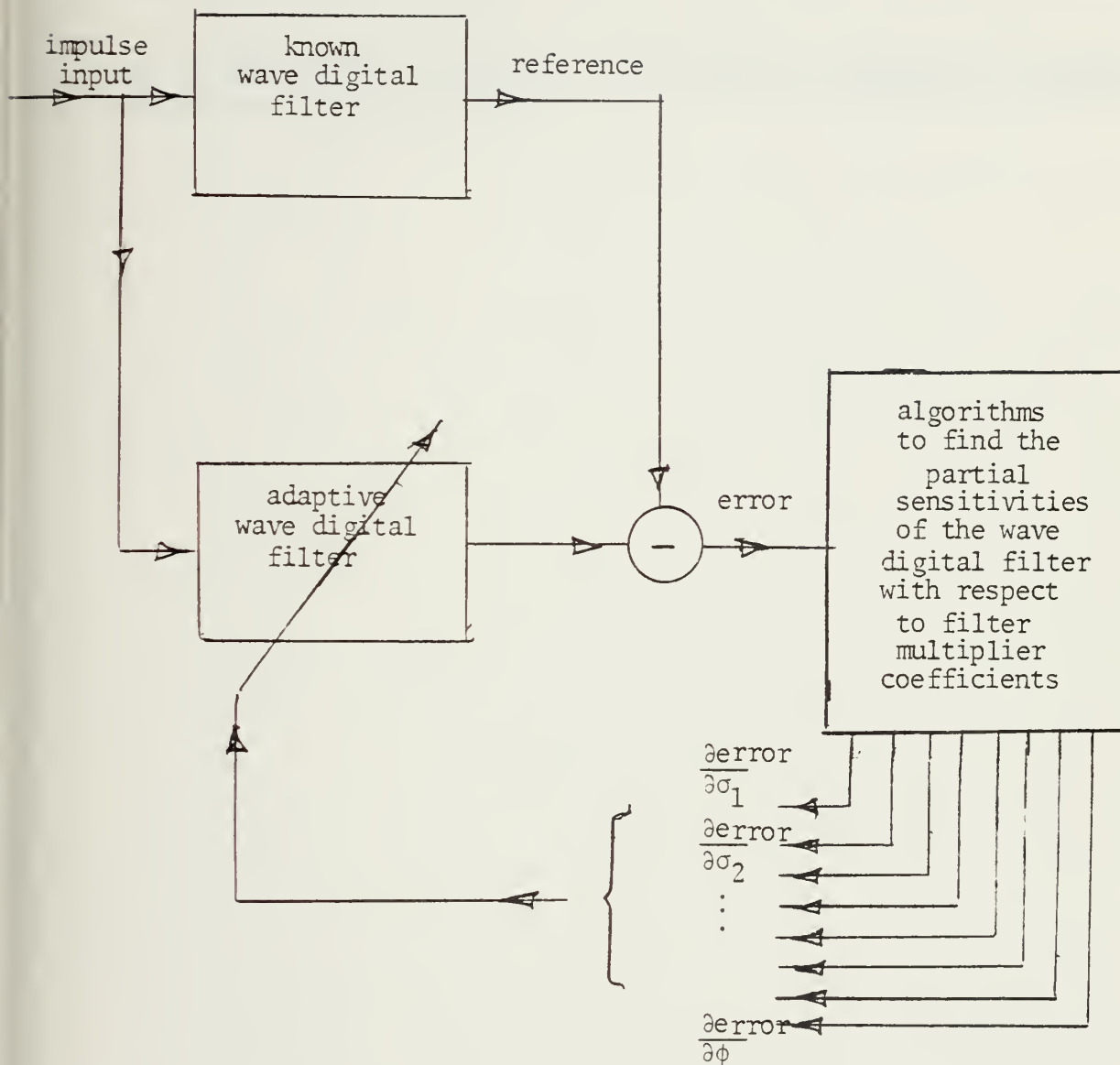


Fig. 7.5. A schematic diagram for the proposed adaptive wave digital filter.



## References

1. S. Erfani, Design of fixed and variable digital filters using generalized delay units, Ph.D. dissertation, 1976, Southern Methodist University, Dallas, Texas.
2. K. S. Thyagarajan, One and two dimensional wave digital filters with low coefficient sensitivities, 1977, Doctoral dissertation, Concordia University, Montreal, Canada.



## APPENDIX 1

### A. DESIGN OF LOW PASS LC FILTER WITH THE GIVEN SPECIFICATION

#### 1. Specification

It is required to design a .5 db low pass Chebechev filter with load resistance  $R_L = 50 \Omega$  and source resistance  $R_S = 100 \Omega$ , with critical frequency of 100 radian/sec.

#### 2. Data

From the Handbook of the Filter Synthesis by Zverev [1] for the given specification, the normalized values of L and C for  $R_L = 1 \Omega$ , making  $R_S = 2.0 \Omega$  as per Figure A.1 are

$$R_S = 2.0 \quad \text{ohms}$$

$$L_1 = .4799 \text{ henries}$$

$$C_2 = .3536 \text{ farads}$$

$$L_3 = 2.2726 \text{ henries}$$

$$C_4 = .7512 \text{ farads}$$

$$L_5 = 3.5532 \text{ henries}$$

$$C_6 = .9513 \text{ farads}$$

$$L_7 = 3.0640 \text{ henries}$$

#### 3. Design of Wave Digital Filter

Using this data the required wave digital filter with no delay free path in port two was designed using the table 4.2b. The schematic wave flow diagram of the wave digital filter is shown in Fig. A.2.

Note that from Figure A.2 and equation (4.17) the unity input impulse response of the filter will be





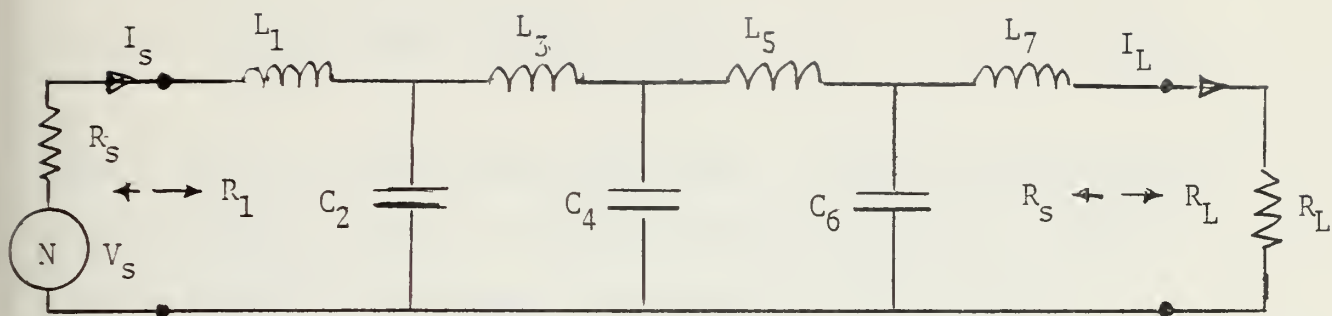


Fig. A.1. Seventh order low pass analogue double resistively terminated LC filter.

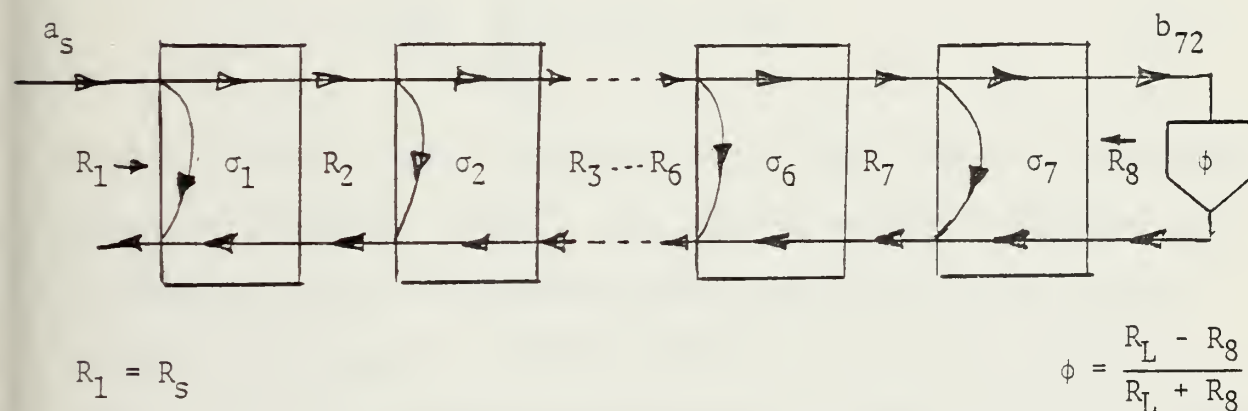


Fig. A.2. Seventh order low pass .5 db ripple Chebyshev wave digital filter designed with no delay free path on port two after the seventh order low pass filter of Fig. A.1. Note that only delay free signal paths are shown.



$$c(nT) = h(nT) = \left(\frac{1+\phi}{2}\right) b_{72}(nT) \quad (A.1)$$

where  $T$  is the sampling period of the filter. We also note that  $h(nT)$  is also the transfer function of the filter in the time domain. Thus the transfer function of the filter in the frequency domain will be

$$H(J\omega) = \sum_{n=0}^N h(nT) e^{-j\omega nT}$$

or

$$\begin{aligned} H(J\omega) &= \sum_{n=0}^N \left(\frac{1+\phi}{2}\right) b_{72}(nT) \cdot e^{-j\omega nT} \\ &= \frac{1+\phi}{2} \sum_{n=0}^N b_{72}(nT) \cdot e^{-j\omega nT} \end{aligned} \quad (A.2)$$

where  $\phi$  is the reflection coefficient of the filter and  $N$  was chosen large enough for transients to decay. Using these results the frequency transfer function and also the impulse response of required filter was programmed in the computer. Sampling period of .01 sec was used, thus making the sampling frequency approximately six times the critical frequency, or three times the Nyquist frequency.

The appropriate computer programs for filter transfer function, and also its impulse response are given at the end of this appendix together with the computer output results which are listed in Tables A.1 and A.2 with their corresponding graphs in Figs. A.4 and A.5.



#### 4. Design of the Conventional Digital Filter for Comparison Purposes

The transfer function of the given filter of Fig. A.1 is of the form

$$H(s) = \frac{V_2(s)}{V_1(s)} \quad (A.3)$$

It can be shown that  $H(s)$  is of the form

$$H(s) = \frac{V_2(s)}{V_1(s)} = \frac{1}{b_7 s^7 + b_6 s^6 + b_5 s^5 + b_4 s^4 + b_3 s^3 + b_2 s^2 + b_1 s + b_0} \quad (A.4)$$

where from the circuit analysis we find

$$b_7 = \frac{L_1 C_2 L_3 C_4 L_5 C_6 L_7}{R_L} \quad (A.5)$$

$$b_6 = C_2 L_3 C_4 L_5 C_6 (L_1 + L_7 \frac{R_S}{R_L})$$

$$b_5 = \frac{L_1}{R_L} [C_2 L_3 (C_4 L_5 + C_4 L_7 + C_6 L_7) + L_5 C_6 L_7 (C_2 + C_4)] \\ + L_3 C_4 L_5 C_6 (C_2 R_S + \frac{L_7}{R_L})$$

$$b_4 = (L_1 + L_7 \frac{R_S}{R_L}) [L_5 C_6 (C_2 + C_4) + C_2 L_3 (C_4 + C_6)] + L_3 C_4 L_5 (C_2 \frac{R_S}{R_L} + C_6)$$

$$b_3 = \frac{C_4}{R_L} (L_1 + L_3) (L_5 + L_7) + (L_3 + L_5) [C_2 C_6 R_S + \frac{L_1 C_2 + C_6 L_7}{R_L}] \\ + C_4 R_S (C_2 L_3 + L_5 C_6) + \frac{L_1 L_7}{R_L} (C_2 + C_6)$$

$$b_2 = (C_4 + C_6) (L_1 + L_3 + L_7 \frac{R_S}{R_L}) + C_2 [L_1 + \frac{R_S}{R_L} (L_3 + L_5 + L_7)] + L_5 (C_4 \frac{R_S}{R_L} + C_6)$$



$$b_1 = \frac{1}{R_L} (L_1 + L_3 + L_5 + L_7) + R_S (C_2 + C_4 + C_6)$$

and

$$b_0 = \frac{R_S + R_L}{R_L}$$

The equation (A.4) can be rewritten in a more familiar form of

$$H(s) = \frac{V_2(s)}{V_1(s)} = \frac{K_1}{s^7 + a_6 s^6 + a_5 s^5 + a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \quad (A.6)$$

where

$$K_1 = \frac{1}{b_7}$$

$$a_6 = \frac{b_6}{b_7}$$

$$a_5 = \frac{b_5}{b_7}$$

etc.

To find the filter scale factor we let  $s \rightarrow 0$ . This leads to the value  $\frac{R_L + R_S}{R_L}$  which is the filter's scale factor.

To find  $T(z)$  for the direct digital filter design, i.e. the digital filter transfer function, we can use equation (A.6) and the bilinear transform equation (2.4) to get

$$T(z) = H(s) \bigg|_{s = \frac{2}{T} \left( \frac{1-z^{-1}}{1+z^{-1}} \right)} \quad (A.7)$$

or for simplicity we can factor  $H(s)$  into one first order and three second order sections as shown in equation (A.8)





$$H(s) = \frac{K}{(s^2 + A_1 s + B_1)(s^2 + A_2 s + B_2)(s^2 + A_3 s + B_3)(s + A_4)} \quad (A.8)$$

Note that in effect we are going to have one first order filter cascaded with three second order cascaded filters. It is much easier to bilinear transform the subsections one at a time rather than bilinear transform the whole  $H(s)$ . Thus each second order section becomes

$$\begin{aligned} \left. \frac{1}{s^2 + A_1 s + B_1} \right|_{s = \frac{2}{T} \left( \frac{1-z^{-1}}{1+z^{-1}} \right)} \\ = K_2 \frac{1 + 2z^{-1} + z^{-2}}{1 + \alpha_1 z^{-1} + \beta_1 z^{-2}} \end{aligned}$$

where

$$\begin{aligned} K_2 &= \frac{1}{\frac{4}{T^2} + \frac{2A_1}{T} + B_1} \\ \alpha_1 &= -2 \frac{\frac{4}{T^2} - B_1}{\frac{4}{T^2} + \frac{2A_1}{T} + B_1} \\ \beta_1 &= \frac{\frac{4}{T^2} - \frac{2A_1}{T} + B_1}{\frac{4}{T^2} + \frac{2A_1}{T} + B_1} \end{aligned}$$

and the first order section transforms into

$$\left. \frac{1}{s + A_4} \right|_{s = \frac{2}{T} \left( \frac{1-z^{-1}}{1+z^{-1}} \right)}$$



$$= K_5 \frac{1 + z^{-1}}{1 + \alpha_4 z^{-1}}$$

where

$$K_5 = \frac{1}{\frac{2}{T} + A_4}$$

$$\alpha_4 = -\frac{\frac{2}{T} - A_4}{\frac{2}{T} + A_4}$$

Thus (A.7) and (A.8) lead to

$$T(z) = K \frac{(1+z^{-1})^2}{(1+\alpha_1 z^{-1} + \beta_1 z^{-2})} \cdot \frac{(1+z^{-1})^2}{(1+\alpha_2 z^{-1} + \beta_2 z^{-2})} \cdot \frac{(1+z^{-1})^2}{(1+\alpha_3 z^{-1} + \beta_3 z^{-2})} \cdot \frac{1+z^{-1}}{1+\alpha_4 z^{-1}}$$

where

$$K = K_1 \cdot K_2 \cdot K_3 \cdot K_4 \cdot K_5$$

Note that  $T(z)$  is merely the transfer function of four cascaded first order and second order sections as shown in Figure A.3. Thus the appropriate iterative equations are

$$\begin{aligned} V_1(nT) &= KV_{in}(nT) + 2KV_{in}(nT-T) + KV_{in}(nT-2T) - \alpha_1 V_1(nT-T) - \beta_1 V_1(nT-2T) \\ V_2(nT) &= V_1(nT) + 2V_1(nT-T) + V_1(nT-2T) - \alpha_2 V_2(nT-T) - \beta_2 V_2(nT-2T) \\ V_3(nT) &= V_2(nT) + 2V_2(nT-T) + V_2(nT-2T) - \alpha_3 V_3(nT-T) - \beta_3 V_3(nT-2T) \\ V_4(nT) &= V_3(nT) + V_3(nT-T) - \alpha_4 V_4(nT-T) \end{aligned} \quad (A.9)$$

where  $T$  is the sampling time of the filter and with initial conditions



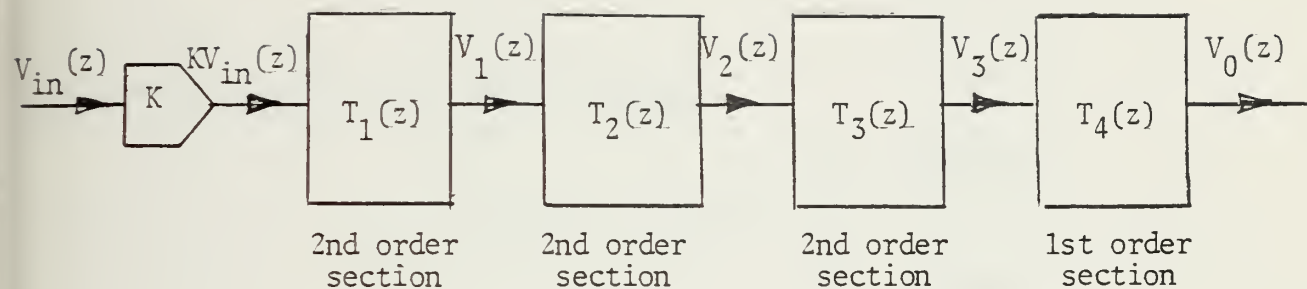


Fig. A.3. Seventh order digitized and cascaded filter corresponding to seventh order analogue filter of Fig. A.1.



set to zero, i.e.

$$\begin{aligned}V_{in}(I) &= 0 & I &= -1, -2, \quad J = 1, 2, 3 \\V_J(I) &= 0 \\V_0(-1) &= 0\end{aligned}$$

Also we note that for impulse response

$$V_{in}(I) = V_{in}(nT) = \begin{cases} 1 & \text{for } n=0 \\ 0 & \text{for } n \neq 0 \end{cases}$$

Thus the unity input impulse response of the filter from equation (A.9) will be  $V_0(nT)$  and we note that this is also the filter transfer function in the time domain, i.e.

$$V_0(nT) = h(nT)$$

Also the filter transfer function in the frequency domain can be found from equation (A.2). Using these results the transfer function and also unity impulse response of filter was programmed in the computer and computed using the same sampling frequency as that of the wave digital filter of Section 3.

The simulation computer programs are given at the end of this appendix and their corresponding outputs are given in Tables A.1, A.2 together with the results obtained for the same filter using wave digital filter design for comparison purposes. Note that no graphs are given for this simulation, since they were exactly the same as the wave digital filter graphs.





## Frequency

## Wave Digital Filter

## Conventional Digital Filter

0.0	01	0.100000	01	0.100000	01
0.200000	01	0.999999	00	0.999999	00
0.400000	01	0.999999	00	0.999999	00
0.600000	01	0.999999	00	0.999999	00
0.800000	01	0.999999	00	0.999999	00
0.100000	02	0.977750	00	0.984800	00
0.120000	02	0.970310	00	0.977750	00
0.140000	02	0.963000	00	0.970320	00
0.160000	02	0.955690	00	0.963000	00
0.180000	02	0.948380	00	0.955690	00
0.200000	02	0.941070	00	0.948380	00
0.220000	02	0.933760	00	0.941070	00
0.240000	02	0.926450	00	0.933760	00
0.260000	02	0.919140	00	0.926450	00
0.280000	02	0.911830	00	0.919140	00
0.300000	02	0.904520	00	0.911830	00
0.320000	02	0.897210	00	0.904520	00
0.340000	02	0.889900	00	0.897210	00
0.360000	02	0.882590	00	0.889900	00
0.380000	02	0.875280	00	0.882590	00
0.400000	02	0.867970	00	0.875280	00
0.420000	02	0.860660	00	0.867970	00
0.440000	02	0.853350	00	0.860660	00
0.460000	02	0.846040	00	0.853350	00
0.480000	02	0.838730	00	0.846040	00
0.500000	02	0.831420	00	0.838730	00
0.520000	02	0.824110	00	0.831420	00
0.540000	02	0.816800	00	0.824110	00
0.560000	02	0.809490	00	0.816800	00
0.580000	02	0.802180	00	0.809490	00
0.600000	02	0.794870	00	0.802180	00
0.620000	02	0.787560	00	0.794870	00
0.640000	02	0.780250	00	0.787560	00
0.660000	02	0.772940	00	0.780250	00
0.680000	02	0.765630	00	0.772940	00
0.700000	02	0.758320	00	0.765630	00
0.720000	02	0.751010	00	0.758320	00
0.740000	02	0.743700	00	0.751010	00
0.760000	02	0.736390	00	0.743700	00
0.780000	02	0.729080	00	0.736390	00
0.800000	02	0.721770	00	0.729080	00
0.820000	02	0.714460	00	0.721770	00
0.840000	02	0.707150	00	0.714460	00
0.860000	02	0.700000	00	0.707150	00
0.880000	02	0.692850	00	0.700000	00
0.900000	02	0.685700	00	0.692850	00
0.920000	02	0.678550	00	0.685700	00
0.940000	02	0.671400	00	0.678550	00
0.960000	02	0.664250	00	0.671400	00
0.980000	02	0.657100	00	0.664250	00
1.000000	02	0.650000	00	0.657100	00
0.100000	03	0.999999	00	0.999999	00
0.120000	03	0.999999	00	0.999999	00
0.140000	03	0.999999	00	0.999999	00
0.160000	03	0.999999	00	0.999999	00
0.180000	03	0.999999	00	0.999999	00
0.200000	03	0.999999	00	0.999999	00
0.220000	03	0.999999	00	0.999999	00
0.240000	03	0.999999	00	0.999999	00
0.260000	03	0.999999	00	0.999999	00
0.280000	03	0.999999	00	0.999999	00
0.300000	03	0.999999	00	0.999999	00
0.320000	03	0.999999	00	0.999999	00
0.340000	03	0.999999	00	0.999999	00
0.360000	03	0.999999	00	0.999999	00
0.380000	03	0.999999	00	0.999999	00
0.400000	03	0.999999	00	0.999999	00
0.420000	03	0.999999	00	0.999999	00
0.440000	03	0.999999	00	0.999999	00
0.460000	03	0.999999	00	0.999999	00
0.480000	03	0.999999	00	0.999999	00
0.500000	03	0.999999	00	0.999999	00
0.520000	03	0.999999	00	0.999999	00
0.540000	03	0.999999	00	0.999999	00
0.560000	03	0.999999	00	0.999999	00
0.580000	03	0.999999	00	0.999999	00
0.600000	03	0.999999	00	0.999999	00
0.620000	03	0.999999	00	0.999999	00
0.640000	03	0.999999	00	0.999999	00
0.660000	03	0.999999	00	0.999999	00
0.680000	03	0.999999	00	0.999999	00
0.700000	03	0.999999	00	0.999999	00
0.720000	03	0.999999	00	0.999999	00
0.740000	03	0.999999	00	0.999999	00
0.760000	03	0.999999	00	0.999999	00
0.780000	03	0.999999	00	0.999999	00
0.800000	03	0.999999	00	0.999999	00
0.820000	03	0.999999	00	0.999999	00
0.840000	03	0.999999	00	0.999999	00
0.860000	03	0.999999	00	0.999999	00
0.880000	03	0.999999	00	0.999999	00
0.900000	03	0.999999	00	0.999999	00
0.920000	03	0.999999	00	0.999999	00
0.940000	03	0.999999	00	0.999999	00
0.960000	03	0.999999	00	0.999999	00
0.980000	03	0.999999	00	0.999999	00
1.000000	03	0.999999	00	0.999999	00

Table A.1. Computer frequency response output for both wave digital filter and conventional cascaded digital filter.



## Time

## Wave Digital Filter

## Conventional Digital Filter

0.0	0.192800-03	0.192800-03
0.100000-01	0.220000-02	0.220000-02
0.200000-01	0.110000-01	0.110000-01
0.300000-01	0.411100-01	0.411100-01
0.400000-01	0.100000-00	0.100000-00
0.500000-01	0.180000-00	0.180000-00
0.600000-01	0.250000-00	0.250000-00
0.700000-01	0.270000-00	0.270000-00
0.800000-01	0.210000-00	0.210000-00
0.900000-01	0.750000-01	0.750000-01
1.000000-00	-0.580000-01	-0.580000-01
1.100000-00	-0.120000-00	-0.120000-00
1.200000-00	-0.100000-00	-0.100000-00
1.300000-00	-0.100000-01	-0.100000-01
1.400000-00	0.040000-01	0.040000-01
1.500000-00	0.840000-01	0.840000-01
1.600000-00	0.380000-01	0.380000-01
1.700000-00	-0.250000-01	-0.250000-01
1.800000-00	-0.570000-01	-0.570000-01
1.900000-00	-0.350000-01	-0.350000-01
2.000000-00	0.140000-01	0.140000-01
2.100000-00	0.490000-01	0.490000-01
2.200000-00	0.420000-01	0.420000-01
2.300000-00	0.580000-02	0.580000-02
2.400000-00	-0.260000-01	-0.260000-01
2.500000-00	-0.340000-01	-0.340000-01
2.600000-00	-0.110000-01	-0.110000-01
2.700000-00	0.160000-01	0.160000-01
2.800000-00	0.250000-01	0.250000-01
2.900000-00	-0.100000-01	-0.100000-01
3.000000-00	-0.110000-01	-0.110000-01
3.100000-00	-0.200000-01	-0.200000-01
3.200000-00	-0.100000-01	-0.100000-01
3.300000-00	0.870000-02	0.870000-02
3.400000-00	0.160000-01	0.160000-01
3.500000-00	-0.100000-01	-0.100000-01
3.600000-00	-0.550000-02	-0.550000-02
3.700000-00	-0.150000-01	-0.150000-01
3.800000-00	-0.110000-01	-0.110000-01
3.900000-00	0.240000-02	0.240000-02
4.000000-00	0.130000-01	0.130000-01
4.100000-00	0.120000-01	0.120000-01
4.200000-00	0.380000-03	0.380000-03
4.300000-00	-0.100000-01	-0.100000-01
4.400000-00	-0.120000-01	-0.120000-01
4.500000-00	-0.280000-02	-0.280000-02
4.600000-00	0.800000-02	0.800000-02
4.700000-00	0.110000-01	0.110000-01
4.800000-00	-0.470000-02	-0.470000-02
4.900000-00	-0.520000-02	-0.520000-02
5.000000-00	-0.590000-02	-0.590000-02
5.100000-00	-0.580000-02	-0.580000-02
5.200000-00	0.270000-02	0.270000-02
5.300000-00	0.820000-02	0.820000-02
5.400000-00	0.630000-02	0.630000-02
5.500000-00	-0.810000-03	-0.810000-03
5.600000-00	-0.620000-02	-0.620000-02
5.700000-00	-0.620000-03	-0.620000-03
5.800000-00	0.460000-02	0.460000-02
5.900000-00	0.560000-02	0.560000-02
6.000000-00	-0.170000-02	-0.170000-02
6.100000-00	-0.350000-02	-0.350000-02
6.200000-00	-0.320000-02	-0.320000-02
6.300000-00	0.230000-02	0.230000-02
6.400000-00	0.220000-02	0.220000-02
6.500000-00	0.450000-02	0.450000-02
6.600000-00	0.270000-02	0.270000-02
6.700000-00	-0.200000-02	-0.200000-02
6.800000-00	-0.180000-02	-0.180000-02
6.900000-00	-0.290000-02	-0.290000-02
7.000000-00	0.340000-03	0.340000-03
7.100000-00	0.300000-02	0.300000-02
7.200000-00	0.290000-02	0.290000-02
7.300000-00	0.310000-03	0.310000-03
7.400000-00	-0.230000-02	-0.230000-02
7.500000-00	-0.270000-02	-0.270000-02
7.600000-00	-0.800000-03	-0.800000-03
7.700000-00	0.160000-02	0.160000-02
7.800000-00	0.240000-02	0.240000-02
7.900000-00	0.110000-02	0.110000-02
8.000000-00	-0.100000-02	-0.100000-02
8.100000-00	-0.210000-02	-0.210000-02
8.200000-00	-0.130000-02	-0.130000-02
8.300000-00	0.550000-03	0.550000-03
8.400000-00	0.170000-02	0.170000-02
8.500000-00	0.130000-02	0.130000-02
8.600000-00	-0.150000-03	-0.150000-03
8.700000-00	-0.140000-02	-0.140000-02
8.800000-00	-0.130000-02	-0.130000-02
8.900000-00	-0.160000-03	-0.160000-03
9.000000-00	0.100000-02	0.100000-02
9.100000-00	0.120000-02	0.120000-02
9.200000-00	0.390000-03	0.390000-03
9.300000-00	-0.700000-03	-0.700000-03
9.400000-00	-0.110000-02	-0.110000-02
9.500000-00	-0.500000-03	-0.500000-03
9.600000-00	0.480000-03	0.480000-03
9.700000-00		

Table A.2. Computer output for unity impulse response of both wave digital filter and conventional cascaded digital filter in the time domain.



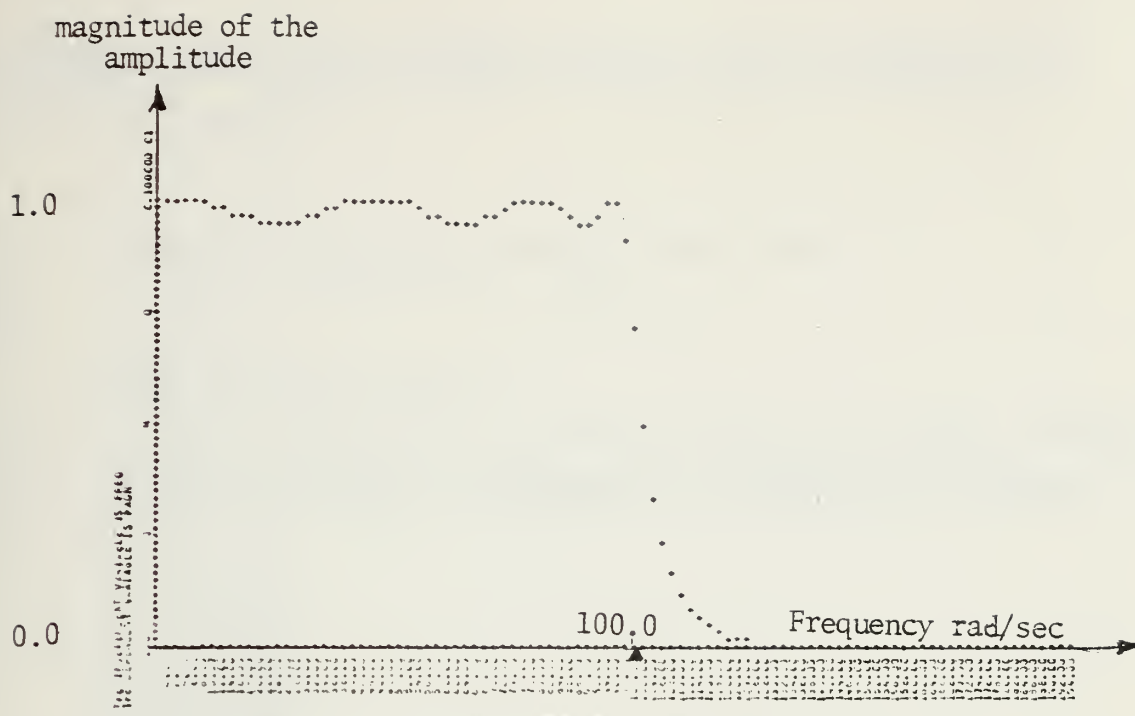


Fig. A.4. Graph of the transfer function of the wave digital filter with the given specification in Section 1.

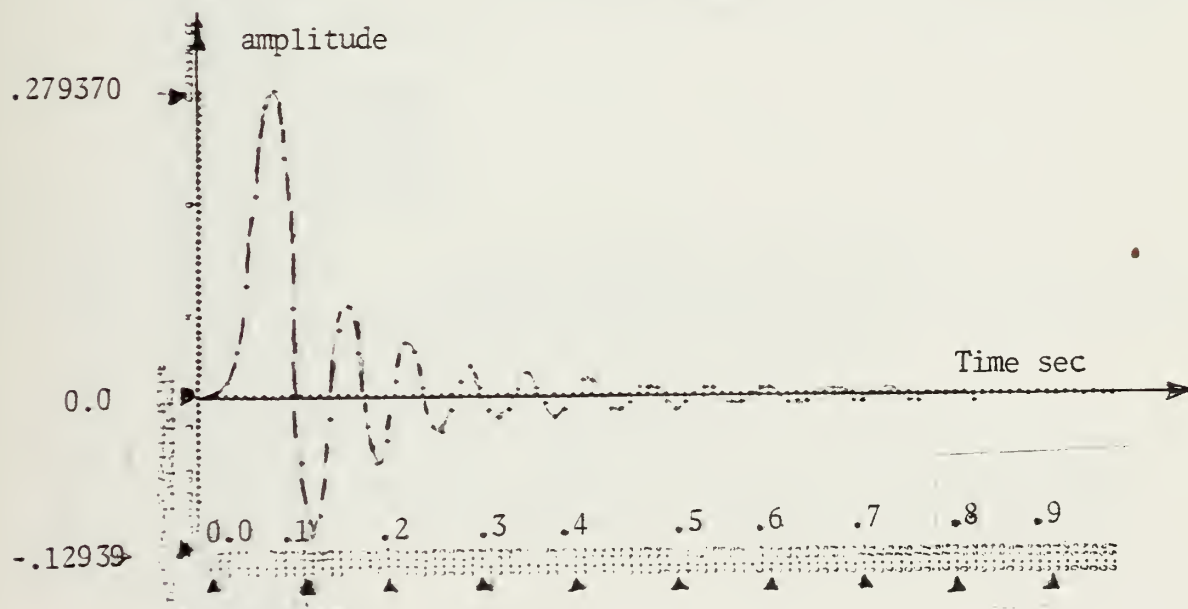


Fig. A.5. Graph of the unity impulse response of wave digital filter with the given specification in Section 1.



- B. Computer Program No. 1. Program for unity impulse response of the wave digital filter with no delay free path on port two with the given specification .

```

C
C ***** WAVE DIGITAL FILTER *****
C
C ***** TIME RESPONSE *****
C
C IMPLICIT REAL*8 (A-H,O-Z)
C DIMENSION DATA(210,3)
C
C *****.500 CHEBECHEV LOW-PASS FILTER WITH RS=2.0 *****
C COMPONENT VALUES
C INDUCTANCE AND CAPACITANCE VALUES IN HENRIES, AND FARADS
C NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C WITH RL=1
C
C RS=2.000
C AL1=2.427500
C C2=.747000
C AL3=4.369500
C C4=.837700
C AL5=4.483600
C C6=.813700
C AL7=3.405000
C RL=1.000
C
27 WRITE(6,27)
C FORMAT(5X,'THE UNSCALED COMPONENT VALUES ARE',//)
C WRITE(6,18) RL,RS
C WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
C
C SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
C OMGAC=100.000
C
C SAMPLING TIME IS T
C T=.0100
C
C IMPEDANCE SCALE FACTOR IS SIMP
C SIMP=50.000
C THE EFFECT OF SAMPLING TIME IS SFREQ
C FREQUENCY SCALING WITH PREWARPING AS WELL AS TAKING INTO ACCOUNT
C SFREQ=DTAN(OMGAC*T/2)
C RS=RS*SIMP
C AL1=AL1*SIMP/SFREQ
C C2=C2/(SIMP*SFREQ)
C AL3=AL3*SIMP/SFREQ
C C4=C4/(SIMP*SFREQ)
C AL5=AL5*SIMP/SFREQ
C C6=C6/(SIMP*SFREQ)
C AL7=AL7*SIMP/SFREQ
C RL=RL*SIMP
C
29 WRITE(6,29)
C FORMAT(5X,'THE SCALED COMPONENT VALUES ARE',//)
22 WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
C
22 FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
C
18 WRITE(6,18) RL,RS
C
18 FORMAT(10X,'RL=',E12.5,5X,'RS=',E12.5,//)
C
C FILTER SCALE FACTOR FROM DATA IS COEF
C COEF=(RS+RL)/RL
37 WRITE(6,37) COEF
C
37 FORMAT(5X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
C
C CALCULATING THE WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
C -----
C
C R1=RS
C R2=R1+AL1
C SIGMA1=R1/R2
C G2=1.000/R2
C G3=G2+C2
C SIGMA2=G2/G3
C R3=1.000/G3
C R4=R3+AL3

```





```

SIGMA3=R3/R4
G4=1.000/R4
G5=G4+C4
SIGMA4=G4/G5
R5=1.000/G5
R6=R5+AL5
SIGMA5=R5/R6
G6=1.000/R6
G7=G6+C6
SIGMA6=G6/G7
R7=1.000/G7
R8=R7+AL7
SIGMA7=R7/R8
PHI=(PL-R8)/(RL+R8)
42  WRITE(6,2)
    FORMAT(5X,'THE WAVE DIGITAL FILTER MULTIPLIER CCEFFICIENTS ARE....
*','/')
10  WRITE(6,10) SIGNAL,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
    FORMAT(1,4X,'SIGMA1=',F6.4,4X,'SIGMA2=',F6.4,4X,'SIGMA3=',F6.4,4X,
*'SIGMA4=',F6.4,4X,'SIGMA5=',F6.4,4X,'SIGMA6=',F6.4,4X,'SIGMA7=',F6
*.4,4X,'PHI=',F6.4)

```

#### INITIAL VALUES

```

INPUT IN TIME DOMAIN IS      AS
AS=1.000
DLTAT=0.000
A11=AS*CDEF
X11=0.000
X12=0.000
X11=0.000
X13=0.000
X14=0.000
X23=0.000
X24=0.000
X33=0.000
X34=0.000
X43=0.000
X44=0.000
X53=0.000
X54=0.000
X63=0.000
X64=0.000
X72=0.000
X73=0.000
X74=0.000

```

#### ITERATION IN THE TIME DOMAIN

```

DC 10C I=1,58
B12=A11+X11-X23+SIGMA1*(X23-X14)
B22=X33+SIGMA2*(B12+X14-X33-X24)
B32=B22+X24-X43+SIGMA3*(X43-X34)
B42=X53+SIGMA4*(B32+X34-X53-X44)
B52=B42+X44-X63+SIGMA5*(X63-X54)
B62=X73+SIGMA6*(B52+X54-X73-X64)
B72=B62+X64-X72+SIGMA7*(X72-X74)
A72=B72*PHI
B71=B62+SIGMA7*(A72-B62+X72-X73)
B61=B71-B52+X73+SIGMA6*(B52-X63)
B51=B42+SIGMA5*(B61-B42+X63-X53)
B41=B51-B32+X53+SIGMA4*(B32-X43)
B31=B22+SIGMA3*(B41-B22+X43-X33)
B21=B31-B12+X33+SIGMA2*(B12-X23)
B11=A11+SIGMA1*(B21-A11+X23-X12)

```

#### UPDATED VALUES

```

X11=A11
X12=B11

```



```

X14=B12
X23=B21
X24=B22
X33=B31
X34=B32
X43=B41
X44=B42
X53=B51
X54=B52
X63=B61
X64=B62
X72=A72
X73=B71
X74=B72
C
C   FOR IMPULSE RESPONSE ALL IS SET TO ZERO FOR THE NEXT ITERATION
C   ALL=0.000
C
C   ARRANGING THE OUTPUT DATA
DATA(I,1)=OLTAT
DATA(I,2)=B72*(1.000+PHI)/2
DATA(I,3)=0.000
OLTAT=OLTAT+T
100 CONTINUE
C
C
54  WRITE(6,54)
    FORMAT('1')
    WRITE(6,55)
55  FORMAT(20X,'TIME',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
    WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
20  CALL GRAPHX(DATA,98,4HTIME,9HMAGNITUDE)
    FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C
    STOP
    END

```



# GRAPHX

```

SUBROUTINE GRAPHX (DATA,N,VINDEP,VARDEP)
IMPLICIT REAL*8 (A-H,C-Z)
DIMENSION DATA(210,3),B(121)
DATA CCT/'+'/,STAR/'*'/,BLANK/' '/
300 WRITE(6,300) VINDEP
   FORMAT(1H1,' THE INDEPENDENT VARIABLE IS ',A4)
400 WRITE(6,400) VARDEP
   FORMAT(1H,' THE DEPENDENT VARIABLE IS ',A4)
500 FCFORMAT(' ')
   BIGEST=DATA(1,2)
   SMAL=DATA(1,2)
   DO 1 I=2,N
     IF (DATA(I,2).GT.BIGEST) BIGEST=DATA(I,2)
     IF (DATA(I,2).LT.SMAL) SMAL=DATA(I,2)
1    CONTINUE
   DO 2 I=1,N
     IF (DATA(I,3).GT.BIGEST) BIGEST=DATA(I,3)
     IF (DATA(I,3).LT.SMAL) SMAL=DATA(I,3)
2    CONTINUE
   IF (SMAL.GE.0.000) SMAL=0.000
   WRITE(6,200) SMAL,BIGEST
   BMIN=BIGEST-SMAL
   DO 3 I=1,62
     B(I)=BLANK
3    DO 4 I=1,N
       DATA(I,2)=(DATA(I,2)-SMAL)*61.000/BMIN+1.000
       DATA(I,3)=(DATA(I,3)-SMAL)*61.000/BMIN+1.000
       INDEX=DATA(I,2)
       JINDEX=DATA(I,3)
       B(INDEX)=DOT
       B(JINDEX)=STAR
       WRITE(6,100) DATA(I,1),(B(NN1),NN1=1,62)
       B(INDEX)=BLANK
       B(JINDEX)=BLANK
4    CONTINUE
200 FCFORMAT(8X,E12.5,5X,'Q',14X,'M',14X,'Q',12X,E12.5,/10X,61(1H*))
100 FORMAT(1X,F8.2,1X,62A1)
RETURN
END

```



- C. Computer Program No. 2. Program for transfer function of the wave digital filter with no delay free path on port two with the given specification.

```

C      ***** WAVE DIGITAL FILTER *****
C      **** FREQUENCY RESPONSE ****
C
C      IMPLICIT REAL*8 (A-H,O-Z)
C      DIMENSION DATA(210,3)
C      COMPLEX*16 H2,W1,Z
C
C      *****.5DB CHEBECHEV LOW-PASS FILTER WITH RS=2.0 *****
C      COMPONENT VALUES
C      INDUCTANCE AND CAPACITANCE VALUES IN HENRIES, AND FARADS
C      NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C      WITH RL=1
C
C      RS=2.0D0
C      AL1=2.4275D0
C      C2=.7470D0
C      AL3=4.3695D0
C      C4=.8377D0
C      AL5=4.4886D0
C      C6=.8137D0
C      AL7=3.4050D0
C      RL=1.0D0
27    WRITE(6,27)
C      FORMAT(5X,'THE UNSCALED COMPONENT VALUES ARE',/)
C      WRITE(6,18) RL,RS
C      WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
C
C      SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
C      CMGAC=100.000
C
C      SAMPLING TIME IS T
C      T=.01D0
C
C      IMPEDANCE SCALE FACTOR IS SIMP
C      SIMP=50.0D0
C
C      FREQUENCY SCALING WITH PREWARPING AS WELL AS TAKING INTO ACCOUNT
C      THE EFFECT OF SAMPLING TIME IS SFREQ
C      SFREQ=DTAN(CMGAC*T/2)
C      RS=RS*SIMP
C      AL1=AL1*SIMP/SFREQ
C      C2=C2/(SIMP*SFREQ)
C      AL3=AL3*SIMP/SFREQ
C      C4=C4/(SIMP*SFREQ)
C      AL5=AL5*SIMP/SFREQ
C      C6=C6/(SIMP*SFREQ)
C      AL7=AL7*SIMP/SFREQ
C      RL=RL*SIMP
29    WRITE(6,29)
C      FORMAT(5X,'THE SCALED COMPONENT VALUES ARE',/)
22    WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
C      FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,3X,'L3=',E12.5,3X,
1    'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,/)
18    WRITE(6,18) RL,RS
C      FORMAT(10X,'RL=',E12.5,5X,'RS=',E12.5,/)
C
C      FILTER SCALE FACTOR FROM DATA IS COEF
C      CCEF=(RS+RL)/RL
37    WRITE(6,37) COEF
C      FORMAT(5X,'THE FILTER SCALE FACTOR IS....',E12.5,/)
C
C      CALCULATING THE WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
C      -----
C      R1=RS
C      R2=R1+AL1
C      SIGMA1=R1/R2
C      G2=1.0D0/R2
C      G3=G2+C2
C      SIGMA2=G2/G3

```

\* Graphx, subroutine is given in Computer Program No. 1.









```

B61=B71-B52+X73+SIGMA6*(B52-X63)
B51=B42+SIGMA5*(B61-B42+X63-X53)
B41=B31-B32+X53+SIGMA4*(B32-X43)
B31=B22+SIGMA3*(B41-B22+X43-X33)
B21=B31-B12+X33+SIGMA2*(B12-X23)
B11=A11+SIGMA1*(B21-A11+X23-X13)

```

```

C
C
C
UPDATED VALUES
-----

```

```

X11=A11
X13=B11
X14=B12
X23=B21
X24=B22
X33=B31
X34=B32
X43=B41
X44=B42
X53=B51
X54=B52
X63=B61
X64=B62
X72=A72
X73=B71
X74=B72
A11=0.000
WT=W*TT
W1=CCVPLX(0.000,-WT)
Z=CDEXP(W1)
H2=H2+B72*Z
TT=TT+T

```

```

100 CONTINUE

```

```

C
C
DB=CDABS(H2)*(1.000+PHI)/2
ARRANGING THE OUTPUT DATA

```

```

DATA(J,1)=W
DATA(J,2)=DB
DATA(J,3)=0.000
W=W+DLTAW
110 CONTINUE

```

```

C
WRITE(6,54)
54 FORMAT('1')
WRITE(6,55)
55 FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
CALL GRAPHX(DATA,98,4HFREQ,SHMAGNITUDE)
20 FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)

```

```

C
STOP
END

```



- D. <sup>\*</sup>Computer Program No. 3. Program for unity impulse response of the conventional cascaded digital filter with the given specification.

```

C *****TIME RESPONSE *****
C
C CONVENTIONAL DIGITAL FILTER DESIGN
C
C IMPLICIT REAL*8 (A-H,O-Z)
C DIMENSION DATA(210,3)
C COMPLEX*16 Z
C DIMENSION A(8),Z(7)
C
C *****.5DB CHEBECHEV LOW-PASS FILTER WITH RS=2.0 *****
C COMPONENT VALUES
C
C INDUCTANCE AND CAPACITANCE VALUES IN HENRIES, AND FARADS
C NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C WITH RL=1
C
C RS=2.000
C AL1=2.427500
C C2=.747000
C AL3=4.369500
C C4=.837700
C AL5=4.488600
C C6=.813700
C AL7=3.405000
C RL=1.000
C WRITE(6,27)
27 FORMAT(5X,'THE UNSCALED COMPONENT VALUES ARE',//)
C WRITE(6,18) RL,RS
C WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
C
C SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
C OMGAC=100.000
C
C SAMPLING TIME IS T
C T=.0100
C
C IMPEDANCE SCALE FACTOR IS SIMP
C SIMP=50.000
C
C FREQUENCY SCALING WITH PREWARPING IS SFREQ
C SFREQ=2*DTAN(OMGAC*T/2)/T
C RS=RS*SIMP
C AL1=AL1*SIMP/SFREQ
C C2=C2/(SIMP*SFREQ)
C AL3=AL3*SIMP/SFREQ
C C4=C4/(SIMP*SFREQ)
C AL5=AL5*SIMP/SFREQ
C C6=C6/(SIMP*SFREQ)
C AL7=AL7*SIMP/SFREQ
C RL=RL*SIMP
C WRITE(6,29)
29 FORMAT(5X,'THE SCALED COMPONENT VALUES ARE',//)
C WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
22 FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
C WRITE(6,18) RL,RS
18 FORMAT(10X,'RL=',E12.5,5X,'RS=',E12.5,//)
C
C CALCULATION OF THE COEFFICIENTS OF
C -----
C
C H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+A1*S1+AC)
C AAC=(RL+RS)/RL
C
C FILTER SCALE FACTOR FROM DATA IS COEF
C COEF=A40
C WRITE(6,37) COEF
37 FORMAT(5X,'THE FILTER SCALE FACTOR IS....',E12.5,//)
C
C AA1=(AL1+AL3+AL5+AL7)/RL+RS*(C2+C4+C6)

```

\* Graphx, subroutine is given in Computer Program No. 1.



```

AA2=(C4+C6)*(AL1+AL3+AL7*RS/RL)+C2*(AL1+(AL3+AL5+AL7)*RS/RL)+AL5*(
*C6+C4*RS/RL)
AA3=(AL1+AL3)*(AL5+AL7)*C4/RL+(AL3+AL5)*(C2*C6*RS+(C6*AL7+C2*AL1)/
*RL)+C4*RS*(C2*AL3+AL5*C6)+AL1*AL7*(C2+C6)/RL
AA4=(AL1+AL7*RS/RL)*(AL5*C6*(C2+C4)+AL5*C2*(C4+C6))+C4*AL5*AL3*(C6
+C2*RS/RL)
AA5=AL1*(C2*AL3*(C4*(AL5+AL7)+C6*AL7)+AL5*C6*AL7*(C2+C4))/RL+AL3*C
*4*AL5*C6*(AL7/RL+C2*RS)
AA6=C2*AL3*C4*AL5*C6*(AL7*RS/RL+AL1)
AA7=(AL1*C2*AL3*C4*AL5*C6*AL7)/RL

```

```

A(1)=1.000
A(2)=AA6/AA7
A(3)=AA5/AA7
A(4)=AA4/AA7
A(5)=AA3/AA7
A(6)=AA2/AA7
A(7)=AA1/AA7
A(8)=AA0/AA7

```

```
COEF1=COEF/AA7
```

```
WRITE(6,41)
```

```
41 FORMAT(5X,'THE COEF.OF H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+
```

```
1A1*S1+A0) ARE',/)
```

```
WRITE(6,40) A(2),A(3),A(4),A(5),A(6),A(7),A(8)
```

```
40 FORMAT(5X,'A6=',E12.5,3X,'A5=',E12.5,3X,'A4=',E12.5,3X,'A3=',E12.5
```

```
1,3X,'A2=',E12.5,3X,'A1=',E12.5,3X,'A0=',E12.5,/) )
```

```
WRITE(6,43) COEF1
```

```
43 FORMAT(5X,'K=',E12.5,//)
```

```
-----
CALCULATION OF THE COEF. OF THE EQN.
-----
```

```
H(S)= K/(S2+A1*S+B1)(S2+A2*S+B2)(S2+A3*S+B3)(S+B4)
```

```
NDEG=7
```

```
CALL ZPOLR(A,NDEG,Z,IER)
```

```
P1=-2.000*REAL(Z(1))
```

```
F1=REAL(Z(1))*2+AIMAG(Z(1))*2
```

```
P2=-2.000*REAL(Z(3))
```

```
F2=REAL(Z(3))*2+AIMAG(Z(3))*2
```

```
P3=-2.000*REAL(Z(5))
```

```
F3=REAL(Z(5))*2+AIMAG(Z(5))*2
```

```
P4=-REAL(Z(7))
```

```
PP=1.000/AA7
```

```
WRITE(6,42)
```

```
42 FORMAT(5X,'THE COEF. OF H(S) ARE.....
1 H(S)=K/(S2+A1*S+B1)(S2+A2*S+B2)(S2+A3*S+B3)(S+B4)',//)
```

```
WRITE(6,60) P1,F1,P2,F2,P3,F3,P4,COEF1
```

```
60 FORMAT(1X,'A1=',E12.5,1X,'B1=',E12.5,2X,'A2=',E12.5,1X,'B2=',E12.5
```

```
1,2X,'A3=',E12.5,1X,'B3=',E12.5,2X,'A4=',E12.5,3X,'K=',E12.5,//)
```

```
-----
CALCULATION OF THE COEF OF H(Z) WITH SAMPLING PERIOD OF T
-----
```

```
H(Z)=K*(1+Z-1)/(1+A4*Z-1)(1+A1*Z-1+B1*Z-2)(1+A2*Z-1+B2*Z-2)(1+A3*Z-1+B3*Z-2)
```

```
F=2.000/T
```

```
FF=F*F
```

```
B1=2.000*(F1-FF)/(FF+P1*F+F1)
```

```
D1=(FF-P1*F+F1)/(FF+P1*F+F1)
```

```
B2=2.000*(F2-FF)/(FF+P2*F+F2)
```

```
D2=(FF-P2*F+F2)/(FF+P2*F+F2)
```

```
B3=2.000*(F3-FF)/(FF+P3*F+F3)
```

```
D3=(FF+P3*F+F3)/(FF+P3*F+F3)
```

```
B4=(P4-F)/(P4+F)
```

```
COEF2=COEF1/((P4+F)*(FF+F*P1+F1)*(FF+F*P2+F2)*(FF+F*P3+F3))
```

```
WRITE(6,66)
```

```
66 FORMAT(5X,'THE COEF. OF Z** -1 AND Z** -2 IN QUADRATURE FORM AND TCT
```

```
1AL MULTPLYING FACTOR K ARE',/)
```

```
WRITE(6,60) B1,D1,B2,D2,B3,D3,B4,COEF2
```





```

C
C
C      INITIAL VALUES
C      -----
C      INPUT IN TIME DOMAIN IS      UP
UP=1.000
UPP=UP*COEF2
X1=0.000
X2=0.000
X3=0.000
X4=0.000
X5=0.000
X6=0.000
X7=0.000
DLTAT=0.000
C
C
C      ITERATION IN THE TIME DOMAIN
C      -----
DO 100 I=1,98
V1=UPP+X1
V2=V1+X2
V3=V2+X3
V=V3+X4
X1=UPP-B4*V1
X2=2*V1-B1*V2+X5
X3=2*V2-B2*V3+X6
X4=2*V3-B3*V+X7
X5=V1-D1*V2
X6=V2-D2*V3
X7=V3-D3*V
C
C      FOR IMPULSE RESPONSE UPP IS SET TO ZERO FOR THE NEXT ITERATION
UPP=0.000
C
C      ARRANGING THE OUTPUT DATA
DATA(I,1)=DLTAT
DATA(I,2)=V
DATA(I,3)=0.000
DLTAT=DLTAT+T
100 CONTINUE
C
C      WRITE(6,54)
54      FORMAT('1')
55      WRITE(6,55)
55      FORMAT(20X,'TIME',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
20      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
20      CALL GRAPHX(DATA,98,4HTIME,9HMAGNITUDE)
20      FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C
C      STOP
END

```



E. <sup>\*</sup>Computer Program No. 4. Program for transfer function of the conventional cascaded digital filter with the given specification.

```

C      **** FREQUENCY RESPONSE ****
C      CONNVENTIONAL DIGITAL FILTER DESIGN
C
C      IMPLICIT REAL*8 (A-H,C-Z)
C      COMPLEX*16 W1,Z,Y,H
C      DIMENSION DATA(210,3)
C      DIMENSION A(8),Y(7)
C
C      COMPONENT VALUES
C      *****.503 CHEBECHEV LOW-PASS FILTER WITH RS=2.0 *****
C      INDUCTANCE AND CAPACITANCE VALUES IN HENRIES, AND FARADS
C      NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C      WITH RL=1
C
C      RS=2.000
C      AL1=2.427500
C      C2=.747000
C      AL3=4.369500
C      C4=.837700
C      AL5=4.488600
C      C6=.813700
C      AL7=3.405000
C      RL=1.000
C      WRITE(6,27)
C      FORMAT(5X,' THE UNSCALED COMPONENT VALUES ARE',//)
C      WRITE(6,18) RL,RS
C      WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
C
C      SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
C      OMGAC=100.000
C
C      SAMPLING TIME IS T
C      T=.0100
C
C      IMPEDANCE SCALE FACTOR IS SIMP
C      SIMP=50.000
C
C      FREQUENCY SCALING WITH PREWARPING IS SFREQ
C      SFREQ=2*DTAN(OMGAC*T/2)/T
C      RS=RS*SIMP
C      AL1=AL1*SIMP/SFREQ
C      C2=C2/(SIMP*SFREQ)
C      AL3=AL3*SIMP/SFREQ
C      C4=C4/(SIMP*SFREQ)
C      AL5=AL5*SIMP/SFREQ
C      C6=C6/(SIMP*SFREQ)
C      AL7=AL7*SIMP/SFREQ
C      RL=RL*SIMP
C      WRITE(6,29)
C      FORMAT(5X,' THE SCALED COMPONENT VALUES ARE',//)
C      WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
C      FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
C      1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,//)
C      WRITE(6,18) RL,RS
C      FORMAT(10X,' RL=',E12.5,5X,'RS=',E12.5,//)
C      18
C      CALCULATION OF THE COEFFICIENTS OF
C      -----
C
C      H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+A1*S1+A0)
C
C      AAC=(RL+RS)/RL
C
C      FILTER SCALE FACTOR FROM DATA IS COEF
C      COEF=AAC
C      WRITE(6,37) COEF
C      37
C      FORMAT(5X,' THE FILTER SCALE FACTOR IS....',E12.5,//)
C
C      AA1=(AL1+AL3+AL5+AL7)/RL+RS*(C2+C4+C6)

```

\* Graphx, subroutine is given in Computer Program No. 1.



```

AA2=(C4+C6)*(AL1+AL3+AL7*RS/RL)+C2*(AL1+(AL3+AL5+AL7)*RS/RL)+AL5*(
*C6+C4*RS/RL)
AA3=(AL1+AL3)*(AL5+AL7)*C4/RL+(AL3+AL5)*(C2*C6*RS+(C6*AL7+C2*AL1)/
*RL)+C4*RS*(C2*AL3+AL5*C6)+AL1*AL7*(C2+C6)/RL
AA4=(AL1+AL7*RS/RL)*(AL5*C6*(C2+C4)+AL3*C2*(C4+C6))+C4*AL5*AL3*(C6
+C2*RS/RL)
AA5=AL1*(C2*AL3*(C4*(AL5+AL7)+C6*AL7)+AL5*C6*AL7*(C2+C4))/RL+AL3*C
*4*AL5*C6*(AL7/RL+C2*RS)
AA6=C2*AL3*C4*AL5*C6*(AL7*RS/RL+AL1)
AA7=(AL1*C2*AL3*C4*AL5*C6*AL7)/RL

```

```

C
A(1)=1.000
A(2)=AA6/AA7
A(3)=AA5/AA7
A(4)=AA4/AA7
A(5)=AA3/AA7
A(6)=AA2/AA7
A(7)=AA1/AA7
A(8)=AA0/AA7

```

```

C
CCEF1=COEF/AA7
WRITE(6,41)
41  FORMAT(5X,'THE COEF. OF H(S)=K/(S7+A6*S6+A5*S5+A4*S4+A3*S3+A2*S2+
1A1*S1+A0) ARE',/)
40  WRITE(6,40) A(2),A(3),A(4),A(5),A(6),A(7),A(8)
FORMAT(5X,'A6=',E12.5,3X,'A5=',E12.5,3X,'A4=',E12.5,3X,'A3=',E12.5
1,3X,'A2=',E12.5,3X,'A1=',E12.5,3X,'A0=',E12.5,/)
43  WRITE(6,43) CCEF1
FORMAT(5X,'K=',E12.5,/)

```

```

CCCC
CALCULATION OF THE COEF. OF THE EQN.

```

$$H(S) = K / (S^2 + A1*S + B1)(S^2 + A2*S + B2)(S^2 + A3*S + B3)(S + B4)$$

```

C
NDEG=7
CALL ZPOLR(A,NDEG,Y,IER)
P1=-2.000*REAL(Y(1))
F1=REAL(Y(1))*2+AIMAG(Y(1))*2
P2=-2.000*REAL(Y(3))
F2=REAL(Y(3))*2+AIMAG(Y(3))*2
P3=-2.000*REAL(Y(5))
F3=REAL(Y(5))*2+AIMAG(Y(5))*2
P4=-REAL(Y(7))
PP=1.000/AA7
42  WRITE(6,42)
FORMAT(5X,'THE COEF. OF H(S) ARE...')
1  H(S)=K/((S2+A1*S+B1)(S2+A2*S+B2)(S2+A3*S+B3)(S+B4)),/)
60  WRITE(6,60) P1,F1,P2,F2,P3,F3,P4,COEF1
FORMAT(1X,'A1=',E12.5,1X,'B1=',E12.5,2X,'A2=',E12.5,1X,'B2=',E12.5
1,2X,'A3=',E12.5,1X,'B3=',E12.5,2X,'A4=',E12.5,3X,'K=',E12.5,/)

```

```

CCCC
CALCULATION OF THE COEF OF H(Z) WITH SAMPLING PERIOD OF T

```

$$H(Z) = K * (1+Z^{-1})^7 / ((1+A4*Z^{-1})(1+A1*Z^{-1}+B1*Z^{-2})(1+A2*Z^{-1}+B2*Z^{-2})(1+A3*Z^{-1}+B3*Z^{-2}))$$

```

C
F=2.000/T
FF=F*F
B1=2.000*(F1-FF)/(FF+P1*F+F1)
D1=(FF-P1*F+F1)/(FF+P1*F+F1)
B2=2.000*(F2-FF)/(FF+P2*F+F2)
D2=(FF-P2*F+F2)/(FF+P2*F+F2)
B3=2.000*(F3-FF)/(FF+P3*F+F3)
D3=(FF-P3*F+F3)/(FF+P3*F+F3)
B4=(P4-F)/(P4+F)
CCEF2=CCEF1/((P4+F)*(FF+F*P1+F1)*(FF+F*P2+F2)*(FF+F*P3+F3))
66  WRITE(6,66)
FORMAT(5X,'THE COEF. OF Z**-1 AND Z**-2 IN QUADRATURE FORM AND TOT
1AL MULTIPLYING FACTOR K ARE...',/)
WRITE(6,60) B1,D1,B2,D2,B3,D3,B4,CCEF2

```



```

C      INPUT IN TIME DOMAIN IS      UP
C      UP=1.000
C      UPP=UP*CDEF2
C
C      FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C      FREQUENCY INCREMENT
C      DLTAW=OMGAC/50.000
C
C      INITIAL VALUES IN FREQUENCY DOMAIN
C      W=0.000
C
C      ITERATION IN THE FREQUENCY DOMAIN
C      -----
C
C      DO 110 J=1,98
C
C      INITIAL VALUES IN TIME DOMAIN
C      UP1=UPP
C      TT=0.000
C      H=CMPLX(0.000,0.000)
C      X1=0.000
C      X2=0.000
C      X3=0.000
C      X4=0.000
C      X5=0.000
C      X6=0.000
C      X7=0.000
C
C      ITERATION IN THE TIME DOMAIN
C      -----
C
C      DO 100 I=1,500
C      V1=UP1+X1
C      V2=V1+X2
C      V3=V2+X3
C      V=V3+X4
C      X1=UP1-B4*V1
C      X2=2*V1-B1*V2+X5
C      X3=2*V2-B2*V3+X6
C      X4=2*V3-B3*V+X7
C      X5=V1-D1*V2
C      X6=V2-D2*V3
C      X7=V3-D3*V
C      UP1=0.000
C      W1=W*IT
C      WT=CMPLX(0.000,-W1)
C      Z=CDEXP(WT)
C      H=H+V*Z
C      IT=TT+T
C100  CONTINUE
C
C      DA=COABS(H)
C      ARRANGING THE OUTPUT DATA
C      DATA(J,1)=W
C      DATA(J,2)=DA
C      DATA(J,3)=0.000
C      W=W+DLTAW
C110  CONTINUE
C
C      WRITE(6,54)
C54   FORMAT('1')
C      WRITE(6,55)
C55   FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
C      WRITE(6,20) ((DATA(M,4),M=1,5),M=1,98)
C      CALL GRAPHX(DATA,98,4*FREQ,4*MAGN)
C20   FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C
C      STOP
C      ENC

```





## Reference

- [1] Anatol I. Zverev, Handbook of Filter Synthesis, 1967, published by John Wiley and Sons, Inc.



## APPENDIX 2

A - Computer Program No. 5. <sup>\*</sup> Program to calculate the rms error due to truncation in the number of bits of wave digital filter multiplier coefficients.

```

C      **** FREQUENCY RESPONSE ****
C
C      ROOT MEAN SQUARE ERROR DUE TO TRUNCATION IN NO. OF BITS *****
C      FOR SIMPLE SECTION WAVE DIGITAL FILTER
C
C      IMPLICIT REAL*8 (A-H,O-Z)
C      DIMENSION DATA(210,3)
C      DIMENSION DATA3(210,3)
C
C      *****.5DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0 *****
C      COMPONENT VALUES
C      INDUCTANCE AND CAPACITANCE VALUES IN HENRIES, AND FARADS
C      NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C      WITH RL=1
C
C      RS=1.000
C      AL1=1.789600
C      C2=1.296100
C      AL3=2.717700
C      C4=1.384800
C      AL5=2.717700
C      C6=1.296100
C      AL7=1.789600
C      RL=1.000
C      WRITE(6,18) RL,RS
18  FORMAT(10X,'RL=',F6.4,5X,'RS=',F6.4,/)
C      WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
22  FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X
1, 'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,/)
C
C      FILTER SCALE FACTOR IS COEF
C      COEF=(RL+RS)/RL
C      WRITE(6,37) COEF
37  FORMAT(7X,'THE FILTER SCALE FACTOR IS....',E12.5,/)
C
C      SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
C      OMGAC=1.000
C
C      SAMPLING PERIOD IS DLTAT
C      DLTAT=1.000
C
C      FREQUENCY SCALING WITH PREWARPING AS WELL AS TAKING INTO ACCOUNT
C      THE EFFECT OF SAMPLING TIME IS SCALE
C      SCALE=1.000/DTAN(OMGAC*DLTAT/2)
C      AL1=AL1*SCALE
C      C2=C2*SCALE
C      AL3=AL3*SCALE
C      C4=C4*SCALE
C      AL5=AL5*SCALE
C      C6=C6*SCALE
C      AL7=AL7*SCALE
C
C      CALCULATION TO FIND THE TERMINATING RESISTANCE OF EACH ELEMENT
C      AND WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
C      WITH NO DELAY FREE PATH ON PORT TWO
C      -----
C      R1=RS
C      R2=R1+AL1
C      SIGMA1=R1/R2
C      G2=1.000/R2
C      G3=G2+C2
C      SIGMA2=G2/G3
C      R3=1.000/G3
C      R4=R3+AL3
C      SIGMA3=R3/R4
C      G4=1.000/R4
C      G5=G4+C4
C      SIGMA4=G4/G5
C      R5=1.000/G5
C      R6=R5+AL5
C      SIGMA5=R5/R6

```

\* Subroutine Graphx is given in Computer Program No. 1 of Appendix 1.



```

G6=1.000/R6
G7=G6+C6
SIGMA6=G6/G7
R7=1.000/G7
R6=R7+AL7
SIGMA7=R7/R8
PHI=(RL-R8)/(RL+R8)
C
42  WRITE(6,42)
    FORMAT(5X,'THE WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS ARE....
    *',//)
C
10  WRITE(6,10) SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
    FORMAT(/,4X,'SIGMA1=',F6.4,4X,'SIGMA2=',F6.4,4X,'SIGMA3=',F6.4,4X,
    1'SIGMA4=',F6.4,4X,'SIGMA5=',F6.4,4X,'SIGMA6=',F6.4,4X,'SIGMA7=',F6
    2.4,4X,'PHI=',F6.4,/)
C
C  INPUT IN TIME DOMAIN IS      AS
AS=1.000
UP=AS*CDEF
C
C  CALCULATION OF THE FREQUENCY RESPONSE OF THE
C  FILTER WITH PRACTICALLY INFINITE PRECISION
C  IN ORDER TO SET THE REFERENCE DATA
C
CALL FILTER(UP,SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PH
*I,DATA,OMGAC,DLTAT)
C
DO 220 JJ=1,20
NOBTS=JJ
C
C  CALCULATION OF THE ERROR IN THE FREQUENCY DOMAIN DUE
C  TO QUANTIZATION OF THE WAVE DIGITAL FILTER PARAMETERS TO THE
C  REQUIRED NO. OF BITS
C
CALL FILERR(UP,SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PH
*I,DATA,NOBTS,ER,OMGAC,DLTAT)
C
C  OUTPUT DATA SET UP
DATA3(JJ,1)=JJ
DATA3(JJ,2)=ER
DATA3(JJ,3)=0.000
C
220 CONTINUE
C
C  PLOT OF THE R.M.S. ERROR DUE TO QUANTIZATION IN THE NO OF
C  BITS OF WAVE DIGITAL FILTER PARAMETERS VERSUS THE NO. OF BITS
C
54  WRITE(6,54)
    FORMAT('1')
55  WRITE(6,55)
    FORMAT(20X,'NO. OF BITS',15X,'MEAN SQUAR ERROR',/)
    WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,20)
20  CALL GRAPHX(DATA3,20,4HNBIT,4HMSER)
    FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C
STOP
END

```



# FILTER

```
SUBROUTINE FILTER(UP, SIGMA1, SIGMA2, SIGMA3, SIGMA4, SIGMA5, SIGMA6, SIG
*MA7, PHI, DATA, DMGAC, DLTAT)
```

```
SUBROUTINE FILTER TAKES THE VALUE OF THE INPUT IN THE TIME
DOMAIN AND PLOTS THE MAGNITUDE VERS. FREQUENCY CURVE
OF THE WAVE DIGITAL FILTER
```

```
IMPLICIT REAL*8 (A-H, O-Z)
COMPLEX*16 H2, W1, Z
DIMENSION DATA(210,3), DT(210,3)
```

```
FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
FREQUENCY INCREMENT
DLW=DMGAC/50
```

```
INITIAL VALUES IN THE FREQUENCY DOMAIN
```

```
W=0.000
```

```
ITERATION IN THE FREQUENCY DOMAIN
```

```
DO 110 J=1,98
```

```
INITIAL VALUES IN TIME DOMAIN
```

```
H2=DCMPLX(0.000,0.000)
```

```
TT=0.000
```

```
X11=0.000
```

```
X13=0.000
```

```
X14=0.000
```

```
X23=0.000
```

```
X24=0.000
```

```
X33=0.000
```

```
X34=0.000
```

```
X43=0.000
```

```
X44=0.000
```

```
X53=0.000
```

```
X54=0.000
```

```
X63=0.000
```

```
X64=0.000
```

```
X72=0.000
```

```
X73=0.000
```

```
X74=0.000
```

```
A11=UP
```

```
ITERATION IN THE TIME DOMAIN
```

```
DO 100 I=1,500
```

```
B12=A11+X11-X23+SIGMA1*(X23-X14)
```

```
B22=X33+SIGMA2*(B12+X14-X33-X24)
```

```
B32=B22+X24-X43+SIGMA3*(X43-X34)
```

```
B42=X53+SIGMA4*(B32+X34-X53-X44)
```

```
B52=B42+X44-X63+SIGMA5*(X63-X54)
```

```
B62=X73+SIGMA6*(B52+X54-X73-X64)
```

```
B72=B62+X64-X72+SIGMA7*(X72-X74)
```

```
A72=B72*PHI
```

```
B71=B62+SIGMA7*(A72-B62+X72-X73)
```

```
B61=B71-B52+X73+SIGMA6*(B52-X63)
```

```
B51=B62+SIGMA5*(B51-B42+X63-X53)
```

```
B41=B51-B32+X53+SIGMA4*(B32-X43)
```

```
B31=B22+SIGMA3*(B41-B22+X43-X33)
```

```
B21=B31-B12+X33+SIGMA2*(B12-X23)
```

```
B11=A11+SIGMA1*(B21-A11+X23-X13)
```

```
UPDATED VALUES
```

```
X11=B11
```

```
X13=B11
```

```
X14=B12
```

```
X23=B21
```

```
X24=B22
```

```
X33=B31
```

```
X34=B32
```

```
X43=B41
```

```
X44=B42
```





```

X53=B51
X54=B52
X63=B61
X64=B62
X72=A72
X73=B71
X74=B72
A11=0.000
WT=X*TT
W1=DCMPLX(0.000,-WT)
Z=CJEXP(W1)
H2=H2+B72*Z
TT=TT+DLTA T
C
100 CONTINUE
DB=CDABS(H2)*(1.000+PHI)/2
C
C ARRANGING THE OUTPUT DATA
DATA(J,1)=W
DATA(J,2)=DB
DATA(J,3)=0.000
C
C ARRANGING THE DATA FOR PLOT SUBROUTINE
DT(J,1)=DATA(J,1)
DT(J,2)=DATA(J,2)
DT(J,3)=DATA(J,3)
C
W=W+DLW
110 CONTINUE
C
C WRITING AND PLOTTING THE FREQUENCY RESPONSE
WRITE(6,54)
54 FORMAT('1')
WRITE(6,55)
55 FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
CALL GRAPHX(DT,98,4HFREQ,4HMAGN)
20 FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C
RETURN
END

```



# FILERR

```
SUBROUTINE FILERR(UP,SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIG
*MA7,PHI,DATA,NOBTS,ER,OMGAC,DLTAT)
```

```
SUBROUTINE FILERR TAKES THE ROOT MEAN SQUARE VALUE OF THE ERROR
BETWEEN UNTRUNCATED AND TRUNCATED VALUE OF THE CUT PUT IN THE
FREQUENCY DOMAIN AND ALSO PLOTS THE FINITE PRECISION
FREQUENCY RESPONSE
```

```
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION DATA2(210,3)
DIMENSION DATA(210,3)
```

```
WRITE(6,54)
```

```
FORMAT('1')
```

```
WRITE(6,21)
```

```
FORMAT(10X,'THE UNTRUNCATED VALUES ARE')
```

```
WRITE(6,10) SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
```

```
FORMAT(/,4X,'SIGMA1=',F6.4,4X,'SIGMA2=',F6.4,4X,'SIGMA3=',F6.4,4X,
*'SIGMA4=',F6.4,4X,'SIGMA5=',F6.4,4X,'SIGMA6=',F6.4,4X,'SIGMA7=',F6
*.4,4X,'PHI=',F6.4)
```

```
A=SIGMA1
```

```
B=SIGMA2
```

```
C=SIGMA3
```

```
D=SIGMA4
```

```
E=SIGMA5
```

```
P=SIGMA6
```

```
G=SIGMA7
```

```
H=PHI
```

```
WRITE(6,33) NOBTS
```

```
FORMAT(10X,'NO OF BITS =',12,/)
```

```
PERFORMING THE TRUNCATION PROCESS TO REQUIRED NO. OF BITS ,
NOBTS
```

```
SIGMA1=TRUNC(A,NOBTS)
```

```
SIGMA2=TRUNC(B,NOBTS)
```

```
SIGMA3=TRUNC(C,NOBTS)
```

```
SIGMA4=TRUNC(D,NOBTS)
```

```
SIGMA5=TRUNC(E,NOBTS)
```

```
SIGMA6=TRUNC(P,NOBTS)
```

```
SIGMA7=TRUNC(G,NOBTS)
```

```
PHI=TRUNC(H,NOBTS)
```

```
WRITE(6,22)
```

```
FORMAT(10X,'THE TRUNCATED VALUES ARE')
```

```
WRITE(6,10) SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
```

```
CALCULATION OF THE FREQUENCY RESPONSE WITH WAVE DIGITAL
```

```
PARAMETERS OF FINITE PRECISION
```

```
CALL FILTER(UP,SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PH
*I,DATA2,OMGAC,DLTAT)
```

```
CALCULATION OF ROOT MEAN SQUARE ERROR
```

```
ER=0.000
```

```
DO 213 KK=1,98
```

```
ER=ER+(DATA(KK,2)-DATA2(KK,2))**2
```

```
CONTINUE
```

```
ER=ER/98.000
```

```
ER=DSQRT(ER)
```

```
SIGMA1=A
```

```
SIGMA2=B
```

```
SIGMA3=C
```

```
SIGMA4=D
```

```
SIGMA5=E
```

```
SIGMA6=P
```

```
SIGMA7=G
```

```
PHI=H
```

```
RETURN
```

```
END
```



# TRUNC

FUNCTION TRUNC (VALUE,NOBTS)

FUNCTION TRUNC CONVERTS THE VALUE INTO BINOMIAL FLOATING  
POINT ARITHMETIC AND THEN TRUNCATES THE VALUE TO THE DESIRED  
NO. OF BITS AND THEN CONVERTS BACK THE VALUE INTO DECIMAL  
FIXED POINT ARITHMETIC  
MAX. NO. OF BITS(NOBTS) MUST NOT EXCEED 30  
AND THE ABSOLUTE VALUE OF 'VALUE' MUST BE LESS THAN 1.0D9  
AND THE ABSOLUTE VALUE OF 'VALUE' MUST BE GREATER THAN 1.0D-15

IMPLICIT REAL\*8 (A-H,O-Z)  
DIMENSION MLTPL(30)  
DIMENSION MLT(30)

A=DABS(VALUE)  
JJ=0  
IF(A.LE.1.0D-15) TRUNC=0.0D0  
IF(A.LE.1.0D-15) RETURN  
DSIGN=VALUE/DABS(VALUE)  
IF(A.GE.1.0D0) GO TO 1  
FRACT=A  
C=0.0D0  
NBITS=NOBTS  
THE MAGNITUDE OF A IS GREATER THAN 1

GO TO 20  
1 N=A  
B=N  
FRACT=A-B  
NONT=0  
JJ=1  
2 CONTINUE  
M=N  
IF(N.EC.0) GO TO 38  
N=N/2  
MM=M-2\*N  
IF(MM.EQ.0) GO TO 23  
NONT=NONT+1  
MLT(NONT)=1  
GO TO 2  
23 NONT=NONT+1  
MLT(NONT)=0  
GO TO 2  
38 CONTINUE  
IF(NOBTS.LE.NONT) GO TO 68  
C=B  
NBITS=NBITS-NONT  
GO TO 20  
68 NP=NONT-NOBTS+1  
C=0.0D0  
DO 16 I=NP,NONT  
C=C+MLT(I)\*2\*\*(I-1)  
16 CONTINUE  
TRUNC=C\*DSIGN  
RETURN  
20 CONTINUE  
DO 6 I=1,NBITS  
FRACT=FRACT\*2  
IF(FRACT.GE.1.0D0) GO TO 7  
MLTPL(I)=0  
IF((JJ+MLTPL(I)).EQ.0) NBITS=NBITS+1  
GO TO 6  
7 CONTINUE  
FRACT=FRACT-1.0D0  
MLTPL(I)=1  
JJ=1  
6 CONTINUE  
RMANT=0.0D0  
DO 8 I=1,NBITS  
RMANT=RMANT+MLTPL(I)\*2.0D0\*\*(-I)  
8 CONTINUE  
TRUNC=DSIGN\*(C+RMANT)  
RETURN  
END



B - Computer Program No. 6. Program to calculate the rms error due to truncation in the number of bits of complex wave digital filter multiplier coefficients.

```

C      **** FREQUENCY RESPONSE ****
C
C      ROOT MEAN SQUARE ERROR DUE TO TRUNCATION IN NO. OF BITS *****
C      FOR COMPLEX WAVE DIGITAL FILTER
C
C      IMPLICIT REAL*8 (A-H,O-Z)
C      DIMENSION DATA(210,3)
C      DIMENSION DATA3(210,3)
C
C      *****.5DB CHEBECHEV LOW-PASS FILTER WITH PS=1.0 *****
C      COMPONENT VALUES
C      INDUCTANCE AND CAPACITANCE VALUES IN HENRIES, AND FARADS
C      NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C      WITH RL=1
C
C      RS=1.000
C      AL1=1.789600
C      C2=1.296100
C      AL3=2.717700
C      C4=1.384800
C      AL5=2.717700
C      C6=1.296100
C      AL7=1.789600
C      RL=1.000
C      WRITE(6,18) RL,RS
18      FORMAT(10X,'RL=',F6.4,5X,'RS=',F6.4,/)
C      WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
22      FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X,
1,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,/)
C
C      FILTER SCALE FACTOR IS COEF
C      COEF=(RL+RS)/RL
C      WRITE(6,37) COEF
37      FORMAT(7X,'THE FILTER SCALE FACTOR IS....',E12.5,/)
C
C      SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
C      OMGAC=1.000
C
C      SAMPLING PERIOD IS OLTAT
C      OLTAT=1.000
C
C      FREQUENCY SCALING WITH PREWARPING AS WELL AS TAKING INTO ACCOUNT
C      THE EFFECT OF SAMPLING TIME IS SCALE
C      SCALE=1.000/DTAN(OMGAC*OLTAT/2)
C      AL1=AL1*SCALE
C      C2=C2*SCALE
C      AL3=AL3*SCALE
C      C4=C4*SCALE
C      AL5=AL5*SCALE
C      C6=C6*SCALE
C      AL7=AL7*SCALE
C
C      CALCULATION TO FIND THE TERMINATING RESISTANCE OF EACH ELEMENT
C      AND COMPLEX WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
C      WITH NO DELAY FREE PATH ON PORT TWO
C      -----
C      R1=RS
C      R2=(R1+AL1)/(1.000+C2*(R1+AL1))
C      R3=(R2+AL3)/(1.000+C4*(R2+AL3))
C      R4=(R3+AL5)/(1.000+C6*(R3+AL5))
C      R5=R4+AL7
C      BETA11=(2*R1+AL1+R1*R1*C2-AL1*AL1*C2)/((R1+AL1)*(1.000+R1*C2+AL1*C
*2))
C      GAMMA11=R1/((R1+AL1)*(1.000+R1*C2+AL1*C2))
C      BETA12=(R1*R1*C2+2*R1*AL1*C2-AL1*AL1*AL1*C2)/((R1+AL1)*(1.000+R1*C
*2+AL1*C2))
C      GAMMA12=(R1*R1*C2-AL1-AL1*AL1*C2)/((R1+AL1)*(1.000+R1*C2+AL1*C2))
C      BETA13=1.000/(1.000+R1*C2+AL1*C2)
C      BETA14=R1/(R1+AL1)
C      BETA21=(2*R2+AL3+R2*R2*C4-AL3*AL3*C4)/((R2+AL3)*(1.000+R2*C4+AL3*C
*4))
C      GAMMA21=R2/((R2+AL3)*(1.000+R2*C4+AL3*C4))
C      BETA22=(R2*R2*C4+2*R2*AL3*C4-AL3*AL3*AL3*C4)/((R2+AL3)*(1.000+R2*C

```

\* Subroutine Graphx is given in Computer Program No. 1 of Appendix 1 and Function Trunc is given in Computer Program No. 5.





```

*4+AL3*C4))
GAMA22=(R2*R2*C4-AL3-AL3*AL3*C4)/((R2+AL3)*(1.000+R2*C4+AL3*C4))
BETA123=1.000/(1.000+R2*C4+AL3*C4)
BETA24=R2/(R2+AL3)
BETA31=(2*R3+AL5+R3*R3*C6-AL5*AL5*C6)/((R3+AL5)*(1.000+R3*C6+AL5*C6))
GAMA31=R3/((R3+AL5)*(1.000+R3*C6+AL5*C6))
BETA32=(R3*R3*C6+2*R3*AL5*C6-AL5*AL5*AL5*C6)/((R3+AL5)*(1.000+R3*C6+AL5*C6))
GAMA32=(R3*R3*C6-AL5-AL5*AL5*C6)/((R3+AL5)*(1.000+R3*C6+AL5*C6))
BETA33=1.000/(1.000+R3*C6+AL5*C6)
BETA34=R3/(R3+AL5)
SIGM41=R4/R5
PHI=(RL-R5)/(RL+R5)

```

```

C
42 WRITE(6,42)
   FORMAT(5X,'THE COMPLEX WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
* ARE...',//)

```

```

C
88 WRITE(6,88) BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14
   FORMAT(1X,'BETA11=',E12.5,1X,'GAMA11=',E12.5,1X,'BETA12=',E12.5,1X
*, 'GAMA12=',E12.5,1X,'BETA13=',E12.5,1X,'BETA14=',E12.5,1X,
*, 'GAMA13=',E12.5,1X,'BETA14=',E12.5,1X, 'GAMA14=',E12.5,1X,
89 WRITE(6,89) BETA21,GAMA21,BETA22,GAMA22,BETA23,BETA24
   FORMAT(1X,'BETA21=',E12.5,1X,'GAMA21=',E12.5,1X,'BETA22=',E12.5,1X
*, 'GAMA22=',E12.5,1X,'BETA23=',E12.5,1X,'BETA24=',E12.5,1X,
*, 'GAMA23=',E12.5,1X,'BETA24=',E12.5,1X, 'GAMA24=',E12.5,1X,
90 WRITE(6,90) BETA31,GAMA31,BETA32,GAMA32,BETA33,BETA34
   FORMAT(1X,'BETA31=',E12.5,1X,'GAMA31=',E12.5,1X,'BETA32=',E12.5,1X
*, 'GAMA32=',E12.5,1X,'BETA33=',E12.5,1X,'BETA34=',E12.5,1X,
*, 'GAMA33=',E12.5,1X,'BETA34=',E12.5,1X, 'GAMA34=',E12.5,1X,
91 WRITE(6,91) SIGM41,PHI
   FORMAT(5X,'SIGM41=',E12.5,5X,'PHI=',E12.5,1X, '//)

```

```

C
C INPUT IN TIME DOMAIN IS AS
AS=1.000
UP=AS*COEF

```

```

C
C CALCULATION OF THE FREQUENCY RESPONSE OF THE
C FILTER WITH PRACTICALLY INFINITE PRECESSION
C IN ORDER TO SET THE REFERENCE DATA

```

```

CALL FILTER(UP,BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14,BETA21,GA
*MA21,BETA22,GAMA22,BETA23,BETA24,BETA31,GAMA31,BETA32,GAMA32,BETA3
*3,BETA34,SIGM41,PHI,DATA,OMGAC,DLTAT)

```

```

C
DC 220 JJ=1,20
NOBTS=JJ

```

```

C
C CALCULATION OF THE ERROR IN THE FREQUENCY DOMAIN DUE
C TO QUANTIZATION OF THE WAVE DIGITAL FILTER PARAMETERS TO THE
C REQUIRED NO. OF BITS

```

```

CALL FILERR(UP,BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14,BETA21,GA
*MA21,BETA22,GAMA22,BETA23,BETA24,BETA31,GAMA31,BETA32,GAMA32,BETA3
*3,BETA34,SIGM41,PHI,DATA,NOBTS,ER,OMGAC,DLTAT)

```

```

C
C OUTPUT DATA SET UP
DATA3(JJ,1)=JJ
DATA3(JJ,2)=ER
DATA3(JJ,3)=0.000

```

```

C 220 CONTINUE

```

```

C
C PLOT OF THE R.M.S. ERROR DUE TO QUANTIZATION IN THE NO OF
C BITS OF WAVE DIGITAL FILTER PARAMETERS VERSUS THE NO. OF BITS

```

```

54 WRITE(6,54)
   FORMAT('1')
55 WRITE(6,55)
   FORMAT(20X,'NO. OF BITS',15X,'MEAN SQUAR ERROR',/)
   WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,20)
20 CALL GRAPHX(DATA3,20,4HNSBIT,4HMSER)
   FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)

```

```

C
STOP
END

```



# FILTER

```
SUBROUTINE FILTER(UP,BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14,BET
#A21,GAMA21,BETA22,GAMA22,BETA23,BETA24,BETA31,GAMA31,BETA32,GAMA32
*,BETA33,BETA34,SIGM41,PHI,DATA,DMGAC,DLTAT)
```

```
SUBROUTINE FILTER TAKES THE VALUE OF THE INPUT IN THE TIME
DOMAIN AND PLOTS THE MAGNITUDE VERS. FREQUENCY CURVE
OF THE COMPLEX WAVE DIGITAL FILTER
```

```
IMPLICIT REAL*8 (A-H,O-Z)
COMPLEX*16 H2,W1,Z
DIMENSION DA=A(210,3),DT(210,3)
```

```
FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
FREQUENCY INCREMENT
DLTAW=DMGAC/50.00
```

```
INITIAL VALUES IN THE FREQUENCY DOMAIN
```

```
W=0.000
ITERATION IN THE FREQUENCY DOMAIN
```

```
DO 110 J=1,98
```

```
INITIAL VALUES IN TIME DOMAIN
```

```
H2=DC*PLX(0.000,0.000)
TT=0.000
X11=0.000
X12=0.000
X13=0.000
X14=0.000
X15=0.000
X16=0.000
X17=0.000
X18=0.000
X21=0.000
X22=0.000
X23=0.000
X24=0.000
X25=0.000
X26=0.000
X27=0.000
X28=0.000
X31=0.000
X32=0.000
X33=0.000
X34=0.000
X35=0.000
X36=0.000
X37=0.000
X38=0.000
X41=0.000
X42=0.000
X43=0.000
X44=0.000
A11=UP
```

```
ITERATION IN THE TIME DOMAIN
```

```
DO 100 I=1,500
B12=BETA13*(A11+2*X11+X12)+BETA12*X13+GAMA12*X14-BETA11*X17-GAMA11
**X18
A21=B12
B22=BETA23*(A21+2*X21+X22)+BETA22*X23+GAMA22*X24-BETA21*X27-GAMA21
**X28
A31=B22
B32=BETA33*(A31+2*X31+X32)+BETA32*X33+GAMA32*X34-BETA31*X37-GAMA31
**X38
A41=B32
B42=A41+X41-X42+SIGM41*(X42-X44)
A42=B42*PHI
B41=A41+SIGM41*(A42-A41+X42-X43)
A32=B41
B31=BETA34*(A32+2*X33+X34)-BETA32*X31-GAMA32*A31-BETA31*X35-GAMA31
```



```

C      **X36
C      UPDATED VALUES
C      A22=B31
      B21=BETA24*(A22+2*X23+X24)-BETA22*X21-GAMA22*A21-BETA21*X25-GAMA21
**X26
      A12=B21
      B11=BETA14*(A12+2*X13+X14)-BETA12*X11-GAMA12*A11-BETA11*X15-GAMA11
C      **X15
      UPDATED EQUATIONS
      X12=X11
      X11=A11
      X14=X13
      X13=A12
      X16=X15
      X15=B11
      X18=X17
      X17=B12
      X22=X21
      X21=A21
      X24=X23
      X23=A22
      X26=X25
      X25=B21
      X28=X27
      X27=B22
      X32=X31
      X31=A31
      X34=X33
      X33=A32
      X36=X35
      X35=B31
      X38=X37
      X37=B32
      X41=A41
      X42=A42
      X43=B41
      X44=B42
      A11=0.000
      WT=W*TT
      W1=DC*PLX(0.000,-WT)
      Z=CDEXP(W1)
      H2=H2+B42*Z
      TT=TT+DLTAT
C      100 CONTINUE
      DB=CDABS(H2)*(1.000+PHI)/2
C      ARRANGING THE OUTPUT DATA
      DATA(J,1)=W
      DATA(J,2)=DB
      DATA(J,3)=0.000
C      ARRANGING THE DATA FOR PLOT SUBROUTINE
      DT(J,1)=DATA(J,1)
      DT(J,2)=DATA(J,2)
      DT(J,3)=DATA(J,3)
C
C      W=W+DLTAW
C      110 CONTINUE
C      WRITING AND PLOTTING THE FREQUENCY RESPONSE
      WRITE(6,54)
      FORMAT('1')
      WRITE(6,55)
      54  FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      55  WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      20  CALL GRAPHX(DT,98,4HFREQ,4HMAGN)
      20  FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C      RETURN
      END

```



# FILERR

```
SUBROUTINE FILERR(UP, BETA11, GAMA11, BETA12, GAMA12, BETA13, BETA14, BETA
#A21, GAMA21, BETA22, GAMA22, BETA23, BETA24, BETA31, GAMA31, BETA32, GAMA32
#, BETA33, BETA34, SIGM41, PHI, DATA, NOBTS, ER, DMGAC, DLTAT)
```

```
SUBROUTINE FILERR TAKES THE ROOT MEAN SQUARE VALUE OF THE ERROR
BETWEEN UNTRUNCATED AND TRUNCATED VALUE OF THE CUT PUT IN THE
FREQUENCY DOMAIN AND ALSO PLOTS THE FINITE PRECISION
FREQUENCY RESPONSE
```

```
IMPLICIT REAL*8 (A-H, O-Z)
DIMENSION DATA2(210,3)
DIMENSION DATA(210,3)
```

```
WRITE(6,54)
54 FORMAT('1')
WRITE(6,21)
21 FORMAT(10X, 'THE UNTRUNCATED VALUES ARE')
WRITE(6,88) BETA11, GAMA11, BETA12, GAMA12, BETA13, BETA14
88 FORMAT(1X, 'BETA11=', E12.5, 1X, 'GAMA11=', E12.5, 1X, 'BETA12=', E12.5, 1X
#, 'GAMA12=', E12.5, 1X, 'BETA13=', E12.5, 1X, 'BETA14=', E12.5, //)
WRITE(6,89) BETA21, GAMA21, BETA22, GAMA22, BETA23, BETA24
89 FORMAT(1X, 'BETA21=', E12.5, 1X, 'GAMA21=', E12.5, 1X, 'BETA22=', E12.5, 1X
#, 'GAMA22=', E12.5, 1X, 'BETA23=', E12.5, 1X, 'BETA24=', E12.5, //)
WRITE(6,90) BETA31, GAMA31, BETA32, GAMA32, BETA33, BETA34
90 FORMAT(1X, 'BETA31=', E12.5, 1X, 'GAMA31=', E12.5, 1X, 'BETA32=', E12.5, 1X
#, 'GAMA32=', E12.5, 1X, 'BETA33=', E12.5, 1X, 'BETA34=', E12.5, //)
WRITE(6,91) SIGM41, PHI
91 FORMAT(5X, 'SIGM41=', E12.5, 5X, 'PHI=', E12.5, //)
```

```
AA1=BETA11
AA2=GAMA11
AA3=BETA12
AA4=GAMA12
AA5=BETA13
AA6=BETA14
BB1=BETA21
BB2=GAMA21
BB3=BETA22
BB4=GAMA22
BB5=BETA23
BB6=BETA24
CC1=BETA31
CC2=GAMA31
CC3=BETA32
CC4=GAMA32
CC5=BETA33
CC6=BETA34
DD1=SIGM41
H=PHI
```

```
WRITE(6,33) NOBTS
33 FORMAT(10X, 'NO OF BITS =', I2, //)
```

```
PERFORMING THE TRUNCATION PROCESS TO REQUIRED NO. OF BITS ,
' NOBTS'
```

```
BETA11=TRUNC(AA1,NOBTS)
GAMA11=TRUNC(AA2,NOBTS)
BETA12=TRUNC(AA3,NOBTS)
GAMA12=TRUNC(AA4,NOBTS)
BETA13=TRUNC(AA5,NOBTS)
BETA14=TRUNC(AA6,NOBTS)
BETA21=TRUNC(BB1,NOBTS)
GAMA21=TRUNC(BB2,NOBTS)
BETA22=TRUNC(BB3,NOBTS)
GAMA22=TRUNC(BB4,NOBTS)
BETA23=TRUNC(BB5,NOBTS)
BETA24=TRUNC(BB6,NOBTS)
BETA31=TRUNC(CC1,NOBTS)
GAMA31=TRUNC(CC2,NOBTS)
BETA32=TRUNC(CC3,NOBTS)
GAMA32=TRUNC(CC4,NOBTS)
BETA33=TRUNC(CC5,NOBTS)
BETA34=TRUNC(CC6,NOBTS)
```





```

SIGM41=TRUNC(DD1,NDBTS)
PHI=TRUNC(H,NDBTS)
WRITE(6,22)
22  FORMAT(10X,'THE TRUNCATED VALUES ARE')
WRITE(6,88) BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14
WRITE(6,89) BETA21,GAMA21,BETA22,GAMA22,BETA23,BETA24
WRITE(6,90) BETA31,GAMA31,BETA32,GAMA32,BETA33,BETA34
WRITE(6,91) SIGM41,PHI
C
C  CALCULATION OF THE FREQUENCY RESPONSE WITH WAVE DIGITAL
C  PARAMETERS OF FINITE PRECESSION
CALL FILTER(UP,BETA11,GAMA11,BETA12,GAMA12,BETA13,BETA14,BETA21,GA
*MA21,BETA22,GAMA22,BETA23,BETA24,BETA31,GAMA31,BETA32,GAMA32,BETA3
*3,BETA34,SIGM41,PHI,DATA2,OMGAC,DLTAT)
C
C  CALCULATION OF ROOT MEAN SQUARE ERROR
ER=0.0D0
DO 213 KK=1,98
ER=ER+(DATA(KK,2)-DATA2(KK,2))**2
213 CONTINUE
ER=ER/98.0D0
ER=DSQRT(ER)
C
BETA11=AA1
GAMA11=AA2
BETA12=AA3
GAMA12=AA4
BETA13=AA5
BETA14=AA6
BETA34=CC6
BETA21=BB1
GAMA21=BB2
BETA22=BB3
GAMA22=BB4
BETA23=BB5
BETA24=BB6
BETA33=CC5
GAMA32=CC4
BETA32=CC3
GAMA31=CC2
BETA31=CC1
SIGM41=DD1
PHI=H
C
RETURN
END

```



C - Computer Program No. 7. \* Program to calculate the rms error due to truncation in the number of bits of reduced parameter complex wave digital filter multiplier coefficients.

```

C      **** FREQUENCY RESPONSE ****
C
C      ROOT MEAN SQUARE ERROR DUE TO TRUNCATION IN NO. OF BITS *****
C      FOR REDUCED PARAMETER COMPLEX WAVE DIGITAL FILTER
C
C      IMPLICIT REAL*8 (A-H,O-Z)
C      DIMENSION DATA(210,3)
C      DIMENSION DATA3(210,3)
C
C      *****.50DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0 *****
C      COMPONENT VALUES
C      INDUCTANCE AND CAPACITANCE VALUES IN HENRIES, AND FARADS
C      NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C      WITH RL=1
C
C      RS=1.000
C      AL1=1.789600
C      C2=1.296100
C      AL3=2.717700
C      C4=1.384800
C      AL5=2.717700
C      C6=1.296100
C      AL7=1.789600
C      RL=1.000
C      WRITE(6,18) RL,RS
18      FORMAT(10X,' RL=',F6.4,5X,'RS=',F6.4,/)
C      WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
22      FORMAT(5X,' L1=',E12.5,3X,'C2=',E12.5,' L3=',E12.5,3X,'C4=',E12.5,3X
1,' L5=',E12.5,3X,'C6=',E12.5,3X,' L7=',E12.5,/)
C
C      FILTER SCALE FACTOR IS COEF
C      COEF=(RL+RS)/RL
C      WRITE(6,37) COEF
37      FORMAT(7X,' THE FILTER SCALE FACTOR IS....',E12.5,/)
C
C      SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
C      OMGAC=1.000
C
C      SAMPLING PERIOD IS DLTAT
C      DLTAT=1.000
C
C      FREQUENCY SCALING WITH PREWARPING AS WELL AS TAKING INTO ACCOUNT
C      THE EFFECT OF SAMPLING TIME IS SCALE
C      SCALE=1.000/DTAN(OMGAC*DLTAT/2)
C      AL1=AL1*SCALE
C      C2=C2*SCALE
C      AL3=AL3*SCALE
C      C4=C4*SCALE
C      AL5=AL5*SCALE
C      C6=C6*SCALE
C      AL7=AL7*SCALE
C
C      CALCULATION TO FIND THE TERMINATING RESISTANCE OF EACH ELEMENT
C      AND REDUCED PARAMETER COMPLEX WAVE DIGITAL FILTER COEFFICIENTS
C      WITH NO DELAY FREE PATH ON PORT TWO
C      -----
C      R1=RS
C      ALFA1=AL1/(R1+AL1)
C      BETA1=1.000/(1.000+R1*C2+AL1*C2)
C      R2=AL1*BETA1/ALFA1
C      ALFA2=AL3/(R2+AL3)
C      BETA2=1.000/(1.000+R2*C4+AL3*C4)
C      R3=AL3*BETA2/ALFA2
C      ALFA3=AL5/(R3+AL5)
C      BETA3=1.000/(1.000+R3*C6+AL5*C6)
C      R4=AL5*BETA3/ALFA3
C      R5=R4+AL7
C      SIGMA1=R4/R5
C      PHI=(RL-R5)/(RL+R5)
C      WRITE(6,42)
42      FORMAT(5X,' THE REDUCED PARAMETER COMPLEX WAVE DIGITAL FILTER COEFF

```

\* Subroutine Graphx is given in Computer Program No. 1 Appendix 1 and Function Trunc is given in Computer Program No. 5.



```

C      *ICIENTS ARE...',//)
C
88      WRITE(6,88) ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3
      FORMAT(1X,' ALFA1=',E12.5,1X,'BETA1=',E12.5,1X,'ALFA2=',E12.5,1X
*, 'BETA2=',E12.5,1X,'ALFA3=',E12.5,1X,'BETA3=',E12.5,//)
91      WRITE(6,91) SIGM41,PHI
      FORMAT(5X,'SIGM41=',E12.5,5X,'PHI=',E12.5,//)
C
C      INPUT IN TIME DOMAIN IS      AS
      AS=1.000
      UP=AS*COEF
C
C      CALCULATION OF THE FREQUENCY RESPONSE OF THE
C      FILTER WITH PRACTICALLY INFINITE PRECISION
C      IN ORDER TO SET THE REFERENCE DATA
C
C      CALL FILTER(UP,ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3,SIGM41,PHI,DATA
*,OMGAC,DLTAT)
C
C      DO 220 JJ=1,20
      NOBITS=JJ
C
C      CALCULATION OF THE ERROR IN THE FREQUENCY DOMAIN DUE
C      TO QUANTIZATION OF THE WAVE DIGITAL FILTER PARAMETERS TO THE
C      REQUIRED NO. OF BITS
C
C      CALL FILERR(UP,ALFA1,BETA1,ALFA2,BETA2,ALFA3,BETA3,SIGM41,PHI,DATA
*,NOBITS,ER,OMGAC,DLTAT)
C
C      OUTPUT DATA SET UP
      DATA3(JJ,1)=JJ
      DATA3(JJ,2)=ER
      DATA3(JJ,3)=0.000
C
C 220  CONTINUE
C
C      PLOT OF THE R.M.S. ERROR DUE TO QUANTIZATION IN THE NO OF
C      BITS OF WAVE DIGITAL FILTER PARAMETERS VERSUS THE NO. OF BITS
C
C      WRITE(6,54)
      FORMAT('11')
      WRITE(6,55)
      FORMAT(20X,'NO. OF BITS',15X,'MEAN SQUAR ERROR',/)
      WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,20)
      CALL GRAPHX(DATA3,20,4HNBIT,4HMSER)
      FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C
C      STOP
      END

```



# FILTER

SUBROUTINE FILTER (UP, ALFA1, BETA1, ALFA2, BETA2, ALFA3, BETA3, SIGM41, PH  
\*1, DATA, OMGAC, DLTAT)

SUBROUTINE FILTER TAKES THE VALUE OF THE INPUT IN THE TIME  
DOMAIN AND PLOTS THE MAGNITUDE VERS. FREQUENCY CURVE  
OF THE REDUCED PARAMETER COMPLEX WAVE DIGITAL FILTER

IMPLICIT REAL\*8 (A-H, O-Z)  
COMPLEX\*16 H2, W1, Z  
DIMENSION DATA(210,3), DT(210,3)

FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY  
FREQUENCY INCREMENT  
DLTAW=OMGAC/50.00

INITIAL VALUES IN THE FREQUENCY DOMAIN  
W=0.000

PREMULTIPLICATIONS  
GAMA1=2\*ALFA1  
GAMA2=2\*ALFA2  
GAMA3=2\*ALFA3  
THETA1=ALFA1\*BETA1  
THETA2=ALFA2\*BETA2  
THETA3=ALFA3\*BETA3

ITERATION IN THE FREQUENCY DOMAIN

DO 110 J=1,98

INITIAL VALUES IN TIME DOMAIN

H2=DCMPLX(0.000,0.000)

TT=0.000

X11=0.000

X12=0.000

X13=0.000

X14=0.000

X15=0.000

X16=0.000

X17=0.000

X18=0.000

X21=0.000

X22=0.000

X23=0.000

X24=0.000

X25=0.000

X26=0.000

X27=0.000

X28=0.000

X31=0.000

X32=0.000

X33=0.000

X34=0.000

X35=0.000

X36=0.000

X37=0.000

X38=0.000

X41=0.000

X42=0.000

X43=0.000

X44=0.000

A11=UP

ITERATION IN THE TIME DOMAIN

DO 100 I=1,500

B12=X13+X14-X17+GAMA1\*(X17-X14)+BETA1\*(A11+2\*X11+X12-X13-X14-X17-X  
\*18)+THETA1\*(X13-X17+X14)

A21=B12

B22=X23+X24-X27+GAMA2\*(X27-X24)+BETA2\*(A21+2\*X21+X22-X23-X24-X27-X  
\*28)+THETA2\*(X23-X27+X24)

A31=B22





```

      B32=X33+X34-X37+GAMA3*(X37-X34)+BETA3*(A31+2*X31+X32-X33-X34-X37-X
      *38)+THETA3*(X38-X33-X37+X34)
      A41=B32
      B42=A41+X41-X42+SIGM41*(X42-X44)
      A42=B42*PHI
      B41=A41+SIGM41*(A42-A41+X42-X43)
      A32=B41
      B31=A32-A31-X31+2*X33+X34-X35+ALFA3*(2*(A31-X33+X35)-A32-X34)+BETA
      *3*(A31+X31-X35-X36)+THETA3*(X31-A31-X35+X36)
      A22=B31
      B21=A22-A21-X21+2*X23+X24-X25+ALFA2*(2*(A21-X23+X25)-A22-X24)+BETA
      *2*(A21+X21-X25-X26)+THETA2*(X21-A21-X25+X26)
      A12=B21
      B11=A12-A11-X11+2*X13+X14-X15+ALFA1*(2*(A11-X13+X15)-A12-X14)+BETA
      *1*(A11+X11-X15-X16)+THETA1*(X11-A11-X15+X16)

```

# UPDATED EQUATIONS

```

      X12=X11
      X11=A11
      X14=X13
      X13=A12
      X16=X15
      X15=B11
      X18=X17
      X17=B12
      X22=X21
      X21=A21
      X24=X23
      X23=A22
      X26=X25
      X25=B21
      X28=X27
      X27=B22
      X32=X31
      X31=A31
      X34=X33
      X33=A32
      X36=X35
      X35=B31
      X38=X37
      X37=B32
      X41=A41
      X42=A42
      X43=B41
      X44=B42
      A11=0.000

```

```

      WT=W*TT
      W1=DCMPLX(0.000,-WT)
      Z=CDEXP(W1)
      H2=H2+B42*Z
      TT=TT+DLTAT

```

```

100 CONTINUE
      DB=CDABS(H2)*(1.000+PHI)/2

```

```

      ARRANGING THE OUTPUT DATA
      DATA(J,1)=W
      DATA(J,2)=DB
      DATA(J,3)=0.000

```

```

      ARRANGING THE DATA FOR PLOT SUBROUTINE
      DT(J,1)=DATA(J,1)
      DT(J,2)=DATA(J,2)
      DT(J,3)=DATA(J,3)

```

```

      W=W+DLTAW
110 CONTINUE

```

```

      WRITING AND PLOTTING THE FREQUENCY RESPONSE
      WRITE(6,54)
      FORMAT('1')
      WRITE(6,55)

```

```

55 FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DT,98,4HFREQ,4HMAGN)
20 FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)

```

```

      RETURN
      END

```



# FILERR

SUBROUTINE FILERR(UP, ALFA1, BETA1, ALFA2, BETA2, ALFA3, BETA3, SIGM41, PH  
#1, DATA, NOBTS, ER, DMGAC, DLTAT)

SUBROUTINE FILERR TAKES THE ROOT MEAN SQUARE VALUE OF THE ERROR  
BETWEEN UNTRUNCATED AND TRUNCATED VALUE OF THE OUT PUT IN THE  
FREQUENCY DOMAIN AND ALSO PLOTS THE FINITE PRECISION  
FREQUENCY RESPONSE

IMPLICIT REAL\*8 (A-H,O-Z)  
DIMENSION DATA2(210,3)  
DIMENSION DATA(210,3)

WRITE(6,54)  
FORMAT('1')  
WRITE(6,21)  
FORMAT(10X, 'THE UNTRUNCATED VALUES ARE')  
WRITE(6,88) ALFA1, BETA1, ALFA2, BETA2, ALFA3, BETA3  
FORMAT(1X, ' ALFA1=', E12.5, 1X, 'BETA1 =', E12.5, 1X, 'ALFA2 =', E12.5, 1X  
\*, 'BETA2 =', E12.5, 1X, 'ALFA3 =', E12.5, 1X, 'BETA3 =', E12.5, //)  
WRITE(6,91) SIGM41, PHI  
FORMAT(5X, 'SIGM41=', E12.5, 5X, 'PHI=', E12.5, //)

AA1=ALFA1  
AA2=BETA1  
AA3=ALFA2  
AA4=BETA2  
AA5=ALFA3  
AA6=BETA3  
DD1=SIGM41  
H=PHI

WRITE(6,33) NOBTS  
FORMAT(10X, ' NO OF BITS =', I2, //)

PERFORMING THE TRUNCATION PROCESS TO REQUIRED NO. OF BITS ,  
' NOBTS'

ALFA1=TRUNC(AA1, NOBTS)  
BETA1=TRUNC(AA2, NOBTS)  
ALFA2=TRUNC(AA3, NOBTS)  
BETA2=TRUNC(AA4, NOBTS)  
ALFA3=TRUNC(AA5, NOBTS)  
BETA3=TRUNC(AA6, NOBTS)  
SIGM41=TRUNC(DD1, NOBTS)  
PHI=TRUNC(H, NOBTS)

WRITE(6,22)  
FORMAT(10X, 'THE TRUNCATED VALUES ARE')  
WRITE(6,88) ALFA1, BETA1, ALFA2, BETA2, ALFA3, BETA3  
WRITE(6,91) SIGM41, PHI

CALCULATION OF THE FREQUENCY RESPONSE WITH WAVE DIGITAL  
PARAMETERS OF FINITE PRECISION  
CALL FILTER(UP, ALFA1, BETA1, ALFA2, BETA2, ALFA3, BETA3, SIGM41, PHI, DATA  
#2, DMGAC, DLTAT)

CALCULATION OF ROOT MEAN SQUARE ERROR  
ER=0.000  
DO 213 KK=1,98  
ER=ER+(DATA(KK,2)-DATA2(KK,2))\*2  
CONTINUE  
ER=ER/98.000  
ER=DSQRT(ER)

ALFA1=AA1  
BETA1=AA2  
ALFA2=AA3  
BETA2=AA4  
ALFA3=AA5  
BETA3=AA6  
SIGM41=DD1  
PHI=H

RETURN  
END



D - Computer Program No. 8. \* Program to calculate the rms error due to truncation in the number of bits of the RLC components of the conventional direct digital filter.

```

C      **** FREQUENCY RESPONSE ****
C
C      ROOT MEAN SQUARE ERROR DUE TO TRUNCATION IN NO. OF BITS
C      FOR CONVENTIONAL DIRECT DIGITAL FILTER DESIGN
C
C      IMPLICIT REAL*8 (A-H,O-Z)
C      DIMENSION DATA(210,3)
C      DIMENSION DATA3(210,3)
C
C      *****.5DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0 *****
C      COMPONENT VALUES
C      INDUCTANCE AND CAPACITANCE VALUES IN HENRIES, AND FARADS
C      NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C      WITH RL=1
C
C      RS=1.000
C      AL1=1.789600
C      C2=1.296100
C      AL3=2.717700
C      C4=1.384800
C      AL5=2.717700
C      C6=1.296100
C      AL7=1.789600
C      RL=1.000
C
C      FILTER SCALE FACTOR IS COEF
C      COEF=(RL+RS)/RL
C      WRITE(6,37) COEF
37  FORMAT(7X,'THE FILTER SCALE FACTOR IS....',E12.5,/)
C      WRITE(6,27)
27  FORMAT(5X,'THE UNSCALED COMPONENT VALUES ARE',/)
C      WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
22  FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,'L3=',E12.5,3X,'C4=',E12.5,3X,
1, 'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,/)
C      WRITE(6,28) RS,RL
28  FORMAT(20X,'RS=',E12.5,5X,'RL=',E12.5,/)
C
C      SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
C      CMGAC=1.000
C
C      SAMPLING TIME IS T
C      T=1.000
C
C      FREQUENCY SCALING WITH PREWARPING IS SFREQ
C      SFREQ=2*DTAN(CMGAC*T/2)/T
C      AL1=AL1/SFREQ
C      C2=C2/SFREQ
C      AL3=AL3/SFREQ
C      C4=C4/SFREQ
C      AL5=AL5/SFREQ
C      C6=C6/SFREQ
C      AL7=AL7/SFREQ
C      WRITE(6,29)
29  FORMAT(5X,'THE SCALED COMPONENT VALUES ARE',/)
C      WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
C
C      INPUT IN TIME DOMAIN IS UP
C      UP=1.000
C
C      CALCULATION OF THE FREQUENCY RESPONSE OF THE
C      FILTER WITH PRACTICALLY INFINITE PRECISION
C      IN ORDER TO SET THE REFERENCE DATA
C
C      CALL FILTER(UP,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,CMGAC,DATA,T)
C
C      DC 220 JJ=1,20
C      NOSTS=JJ
C
C      CALCULATION OF THE ERROR IN THE FREQUENCY DOMAIN DUE
C      TO QUANTIZATION OF THE FILTER COMPONENT VALUES TO THE REQUIRED NO.

```

\* Subroutine Graphx is given in Computer Program No. 1 of Appendix 1 and Function Trunc is given in Computer Program No. 5.



```

C      OF BITS
C      CALL FILERR (UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,NOBTS,ER
*,T)
C      OUTPUT DATA SET UP
      DATA3(JJ,1)=JJ
      DATA3(JJ,2)=ER
      DATA3(JJ,3)=0.000
C 220  CONTINUE
C      PLOT OF THE R.M.S. ERROR DUE TO QUANTIZATION IN THE NO OF
C      BITS OF FILTER COMPONENT VALUES VERSUS THE NO. OF BITS
C
      WRITE(6,54)
      FORMAT('1')
      WRITE(6,55)
      FORMAT(20X,'NO. OF BITS',15X,'MEAN SQUAR ERROR',/)
      WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,20)
      CALL GRAPHX(DATA3,20,4HNBIT,4HMSER)
      FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C
      STOP
      END

```





# FILTER

SUBROUTINE FILTER(UP,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,T)

SUBROUTINE FILTER TAKES THE VALUE OF THE INPUT IN THE TIME DOMAIN AND PLOTS THE MAGNITUDE VERSUS FREQUENCY CURVE OF THE CONVENTIONAL DIRECT DIGITAL FILTER

IMPLICIT REAL\*8 (A-H,O-Z)

COMPLEX\*16 WT,Z,Y,H

DIMENSION DATA(210,3),DT(210,3)

DIMENSION A(8),Y(7)

CALCULATION OF COEFFICIENTS OF THE EQUATION

$$H(S) = K / (S^7 + A_6 S^6 + A_5 S^5 + A_4 S^4 + A_3 S^3 + A_2 S^2 + A_1 S + A_0)$$

$$AA0 = (RL + RS) / RL$$

$$AA1 = (AL1 + AL3 + AL5 + AL7) / RL + RS * (C2 + C4 + C6)$$

$$AA2 = (C4 + C6) * (AL1 + AL3 + AL7 * RS / RL) + C2 * (AL1 + (AL3 + AL5 + AL7) * RS / RL) + AL5 * (C6 + C4 * RS / RL)$$

$$AA3 = (AL1 + AL3) * (AL5 + AL7) * C4 / RL + (AL3 + AL5) * (C2 * C6 * RS + (C6 * AL7 + C2 * AL1) / RL) + C4 * RS * (C2 * AL3 + AL5 * C6) + AL1 * AL7 * (C2 + C6) / RL$$

$$AA4 = (AL1 + AL7 * RS / RL) * (AL5 * C6 * (C2 + C4) + AL3 * C2 * (C4 + C6)) + C4 * AL5 * AL3 * (C6 + C2 * RS / RL)$$

$$AA5 = AL1 * (C2 * AL3 * (C4 * (AL5 + AL7) + C6 * AL7) + AL5 * C6 * AL7 * (C2 + C4)) / RL + AL3 * C4 * AL5 * C6 * (AL7 / RL + C2 * RS)$$

$$AA6 = C2 * AL3 * C4 * AL5 * C6 * (AL7 * RS / RL + AL1)$$

$$AA7 = (AL1 * C2 * AL3 * C4 * AL5 * C6 * AL7) / RL$$

$$A(1) = 1.000$$

$$A(2) = AA6 / AA7$$

$$A(3) = AA5 / AA7$$

$$A(4) = AA4 / AA7$$

$$A(5) = AA3 / AA7$$

$$A(6) = AA2 / AA7$$

$$A(7) = AA1 / AA7$$

$$A(8) = AA0 / AA7$$

$$COEF1 = A(8)$$

WRITE(6,41)

41 FORMAT(5X,'THE COEF. OF H(S)=K/(S<sup>7</sup>+A<sub>6</sub>S<sup>6</sup>+A<sub>5</sub>S<sup>5</sup>+A<sub>4</sub>S<sup>4</sup>+A<sub>3</sub>S<sup>3</sup>+A<sub>2</sub>S<sup>2</sup>+A<sub>1</sub>S+A<sub>0</sub>) ARE',/)

WRITE(6,40) A(2),A(3),A(4),A(5),A(6),A(7),A(8)

40 FORMAT(5X,'A<sub>6</sub>=',E12.5,3X,'A<sub>5</sub>=',E12.5,3X,'A<sub>4</sub>=',E12.5,3X,'A<sub>3</sub>=',E12.5,3X,'A<sub>2</sub>=',E12.5,3X,'A<sub>1</sub>=',E12.5,3X,'A<sub>0</sub>=',E12.5,/) 1,3X,'A<sub>2</sub>=',E12.5,3X,'A<sub>1</sub>=',E12.5,3X,'A<sub>0</sub>=',E12.5,/)

WRITE(6,43) COEF1

43 FORMAT(5X,'K=',E12.5,/) )

CALCULATION OF THE COEF. OF THE EQN.

$$H(Z) = K * (1 + Z) / (Z^7 + A_6 Z^6 + A_5 Z^5 + A_4 Z^4 + A_3 Z^3 + A_2 Z^2 + A_1 Z + A_0)$$

$$F1 = 2.000 / T$$

$$F2 = F1 * F1$$

$$F3 = F2 * F1$$

$$F4 = F3 * F1$$

$$F5 = F4 * F1$$

$$F6 = F5 * F1$$

$$F7 = F6 * F1$$

$$ALFA7 = F7 + F6 * A(2) + F5 * A(3) + F4 * A(4) + F3 * A(5) + F2 * A(6) + F1 * A(7) + A(8)$$

$$ALFA6 = -7 * F7 - 5 * F6 * A(2) - 3 * F5 * A(3) - F4 * A(4) + F3 * A(5) + 3 * F2 * A(6) + 5 * F1 * A(7) + 7 * A(8)$$

$$ALFA5 = 21 * F7 + 9 * F6 * A(2) + F5 * A(3) - 3 * F4 * A(4) - 3 * F3 * A(5) + F2 * A(6) + 9 * F1 * A(7) + 21 * A(8)$$

$$ALFA4 = -35 * F7 - 5 * F6 * A(2) + 5 * F5 * A(3) + 3 * F4 * A(4) - 3 * F3 * A(5) - 5 * F2 * A(6) + 5 * F1 * A(7) + 35 * A(8)$$

$$ALFA3 = 35 * F7 - 5 * F6 * A(2) - 5 * F5 * A(3) + 3 * F4 * A(4) + 3 * F3 * A(5) - 5 * F2 * A(6) - 5 * F1 * A(7) + 35 * A(8)$$

$$ALFA2 = -21 * F7 + 9 * F6 * A(2) - F5 * A(3) - 3 * F4 * A(4) + 3 * F3 * A(5) + F2 * A(6) - 9 * F1 * A(7) + 21 * A(8)$$

$$ALFA1 = 7 * F7 - 5 * F6 * A(2) + 3 * F5 * A(3) - F4 * A(4) - F3 * A(5) + 3 * F2 * A(6) - 5 * F1 * A(7) + 7 * A(8)$$

$$ALFA0 = -F7 + F6 * A(2) - F5 * A(3) + F4 * A(4) - F3 * A(5) + F2 * A(6) - F1 * A(7) + A(8)$$



```

ALFA6=ALFA6/ALFA7
ALFA5=ALFA5/ALFA7
ALFA4=ALFA4/ALFA7
ALFA3=ALFA3/ALFA7
ALFA2=ALFA2/ALFA7
ALFA1=ALFA1/ALFA7
ALFA0=ALFA0/ALFA7
COEF2=COEF1/ALFA7
WRITE(6,38)
38  FORMAT(5X,'THE COEFFICIENTS OF  H(Z)=K(1+Z)7/(Z7+A6Z6+A5Z5+A4Z4+A
*3Z3+A2Z2+A1Z+A0) ARE ',//)
WRITE(6,40) ALFA6,ALFA5,ALFA4,ALFA3,ALFA2,ALFA1,ALFA0
WRITE(6,43) COEF2
C
UPP=UP*COEF2
C
C  FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C  FREQUENCY INCREMENT
C  DLW=OMGAC/50
C
C  INITIAL VALUES IN THE FREQUENCY DOMAIN
C  W=0.000
C
C  ITERATION IN THE FREQUENCY DOMAIN
C  -----
C  DO 110 J=1,98
C
C  INITIAL VALUES IN TIME DOMAIN
C
C  UP1=UPP
C  TT=0.000
C  H=DC*PLX(0.000,0.000)
C  X1=0.0
C  X2=0.0
C  X3=0.0
C  X4=0.0
C  X5=0.0
C  X6=0.0
C  X7=0.0
C
C  ITERATION IN THE TIME DOMAIN
C  -----
C  DO 100 I=1,500
C  V=UP1+X1
C  X1=7*UP1-ALFA6*V+X2
C  X2=21*UP1-ALFA5*V+X3
C  X3=35*UP1-ALFA4*V+X4
C  X4=35*UP1-ALFA3*V+X5
C  X5=21*UP1-ALFA2*V+X6
C  X6=7*UP1-ALFA1*V+X7
C  X7=UP1-ALFA0*V
C  UP1=0.000
C  W1=W*TT
C  WT=DC*PLX(0.000,-W1)
C  Z=CDEXP(WT)
C  H=H+V*Z
C  TT=TT+T
C
C 100 CONTINUE
C  DA=CDABS(H)
C
C  ARRANGING THE OUTPUT DATA
C  DATA(J,1)=W
C  DATA(J,2)=DA
C  DATA(J,3)=0.000
C
C  APRANGING THE DATA FOR PLOT SUBROUTINE
C  DT(J,1)=DATA(J,1)
C  DT(J,2)=DATA(J,2)
C  DT(J,3)=DATA(J,3)
C
C  W=W+DLW
C 110 CONTINUE

```



```

C
C      WRITING AND PLOTTING THE FREQUENCY RESPONSE
      WRITE(6,54)
54     FORMAT('1')
      WRITE(6,55)
55     FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
20     CALL GRAPHX(DT,98,4HFREQ,4HMAGN)
      FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C
      RETURN
      END

```



# FILERR

```

SUBROUTINE FILERR(UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,NO
*BITS,ER,T)

SUBROUTINE FILERR TAKES THE ROOT MEAN SQUARE VALUE OF THE ERROR
BETWEEN UNTRUNCATED AND TRUNCATED VALUE OF THE OUT PUT IN THE
FREQUENCY DOMAIN AND ALSO PLOTS THE FINITE PRECISION FREQUENCY
RESPONCE

IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION DATA2(210,3)
DIMENSION DATA(210,3)

WRITE(6,54)
FORMAT('1')
WRITE(6,21)
FORMAT(10X,'THE UNTRUNCATED VALUES ARE',/)
WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,3X,'L3=',E12.5,3X,'C4=',E12.5
*,5X,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,/)
WRITE(6,28) RS,RL
FORMAT(20X,'RS=',E12.5,5X,'RL=',E12.5,/)

A=RS
B=AL1
C=C2
D=AL3
E=C4
F=AL5
G=C6
H=AL7
P=RL
WRITE(6,33) NOBITS
FORMAT(10X,' NO OF BITS =',I2,/)

PERFORMING THE TRUNCATION PROCESS TO REQUIRED NO. OF BITS ,
'NOBITS'
RS=TRUNC(RS,NOBITS)
AL1=TRUNC(AL1,NOBITS)
C2=TRUNC(C2,NOBITS)
AL3=TRUNC(AL3,NOBITS)
C4=TRUNC(C4,NOBITS)
AL5=TRUNC(AL5,NOBITS)
C6=TRUNC(C6,NOBITS)
AL7=TRUNC(AL7,NOBITS)
RL=TRUNC(RL,NOBITS)
WRITE(6,25)
FORMAT(10X,'THE TRUNCATED VALUES ARE')
WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
WRITE(6,28) RS,RL

CALCULATION OF THE FREQUENCY RESPONCE WITH COMPLENENTS OF
FINITE PRECISION
CALL FILTER(UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA2,T)

CALCULATION OF ROOT MEAN SQUARE ERROR
ER=0.000
DO 213 KK=1,98
ER=ER+(DATA(KK,2)-DATA2(KK,2))*#2
CONTINUE
ER=ER/98.000
ER=DSQRT(ER)

RS=A
AL1=B
C2=C
AL3=D
C4=E
AL5=F
C6=G
AL7=H
RL=P

RETURN
END

```





E - Computer Program No. 9. \* Program to calculate the rms error due to truncation in the number of bits of the RLC components of the conventional cascaded digital filter.

```

C          **** FREQUENCY RESPONCE ****
C
C          ROOT MEAN SQUARE ERROR DUE TO TRUNCATION IN NO. OF BITS
C          FOR CONVENTIONAL CASCADED DIGITAL FILTER
C
C          IMPLICIT REAL*8 (A-Z)
C          DIMENSION DATA(210,3)
C          DIMENSION DATA3(210,3)
C
C          *****.5DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0 *****
C          COMPONENT VALUES
C          INDUCTANCE AND CAPACITANCE VALJES IN HENRIES, AND FARADS
C          NORMALIZED TO CRITICAL FREQUENCY OF 1 RAD/SEC AT 3 DB POINT
C          WITH RL=1
C
C          RS=1.000
C          AL1=1.789600
C          C2=1.296100
C          AL3=2.717700
C          C4=1.384800
C          AL5=2.717700
C          C6=1.296100
C          AL7=1.789600
C          RL=1.000
C
C          FILTER SCALE FACTOR IS COEF
C          COEF=(RL+RS)/RL
C          WRITE(6,37) COEF
37  FORMAT(7X,' THE FILTER SCALE FACTOR IS....',E12.5,/)
C          WRITE(6,27)
27  FORMAT(5X,' THE UNSCALED COMPONENT VALUES ARE',/)
C          WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
22  FORMAT(5X,' L1=',E12.5,3X,' C2=',E12.5,' L3=',E12.5,3X,' C4=',E12.5,3X
1  , ' L5=',E12.5,3X,' C6=',E12.5,3X,' L7=',E12.5,/)
C          WRITE(6,28) RS,RL
28  FORMAT(20X,' RS=',E12.5,5X,' RL=',E12.5,/)
C
C          SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
C          CMGAC=1.000
C
C          SAMPLING TIME IS T
C          T=1.000
C
C          FREQUENCY SCALING WITH PREWARPING IS SFREQ
C          SFREQ=2*DTAN(CMGAC*T/2)/T
C          AL1=AL1/SFREQ
C          C2=C2/SFREQ
C          AL3=AL3/SFREQ
C          C4=C4/SFREQ
C          AL5=AL5/SFREQ
C          C6=C6/SFREQ
C          AL7=AL7/SFREQ
C          WRITE(6,29)
29  FORMAT(5X,' THE SCALED COMPONENT VALUES ARE',/)
C          WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7
C
C          INPUT IN TIME DOMAIN IS UP
C          UP=1.000
C
C          CALCULATION OF THE FREQUENCY RESPONCE OF THE
C          FILTER WITH PRACTICALLY INFINITE PRECESSION
C          IN ORDER TO SET THE REFERENCE DATA
C
C          CALL FILTER(UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,CMGAC,DATA,T)
C
C          DO 22C JJ=1,20
C          NOETS=JJ
C
C          CALCULATION OF THE ERROR IN THE FREQUENCY DOMAIN DUE
C          TO QUANTIZATION OF THE FILTER COMPONENT VALUES TO THE REQUIRED NO.

```

\* Subroutine Graphx is given in Computer Program No. 1 of Appendix 1 and Frunction Trunc is given in the Computer Program No. 5.



```

C      OF BITS
C      CALL FILERR (UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,CMGAC,DATA,NOBTS,ER
*,T)
C      OUTPUT DATA SET UP
C      DATA3(JJ,1)=JJ
C      DATA3(JJ,2)=ER
C      DATA3(JJ,3)=0.000
C 220  CONTINUE
C      PLOT OF THE R.M.S. ERROR DUE TO QUANTIZATION IN THE NO OF
C      BITS OF FILTER COMPONENT VALUES VERSUS THE NO. OF BITS
C
C      WRITE(6,54)
54      FORMAT('1')
C      WRITE(6,55)
55      FORMAT(20X,'NO. OF BITS',15X,'MEAN SQUAR ERROR',/)
C      WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,20)
20      CALL GRAPHX(DATA3,20,4HNRIT,4HMSER)
C      FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C      STOP
C      END

```



# FILTER

SUBROUTINE FILTER(UP ,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,T)

SUBROUTINE FILTER TAKES THE VALUE OF THE INPUT IN THE TIME DOMAIN AND PLOTS THE MAGNITUDE VERS. FREQUENCY CURVE OF THE CONVENTIONAL CASCADED DIGITAL FILTER

IMPLICIT REAL\*8 (A-H,O-Z)  
COMPLEX\*16 WT,Z,Y,H  
DIMENSION DATA(210,3),DT(210,3)  
DIMENSION A(8),Y(7)

CALCULATION OF COEFFICIENTS OF THE EQUATION

$$H(S)=K/(S^7+A_6*S^6+A_5*S^5+A_4*S^4+A_3*S^3+A_2*S^2+A_1*S+A_0)$$

AA0=(RL+RS)/RL  
AA1=(AL1+AL3+AL5+AL7)/RL+RS\*(C2+C4+C6)  
AA2=(C4+C6)\*(AL1+AL3+AL7\*RS/RL)+C2\*(AL1+(AL3+AL5+AL7)\*RS/RL)+AL5\*(C6+C4\*RS/RL)  
AA3=(AL1+AL3)\*(AL5+AL7)\*C4/RL+(AL3+AL5)\*(C2\*C6\*RS+(C6\*AL7+C2\*AL1)\*RL)+C4\*RS\*(C2\*AL3+AL5\*C6)+AL1\*AL7\*(C2+C6)/RL  
AA4=(AL1+AL7\*RS/RL)\*(AL5+C6\*(C2+C4)+AL3\*C2\*(C4+C6))+C4\*AL5\*AL3\*(C6+C2\*RS/RL)  
AA5=AL1\*(C2\*AL3\*(C4\*(AL5+AL7)+C6\*AL7)+AL5\*C6\*AL7\*(C2+C4))/RL+AL3\*C4\*AL5\*C6\*(AL7/RL+C2\*RS)  
AA6=C2\*AL3\*C4\*AL5\*C6\*(AL7\*RS/RL+AL1)  
AA7=(AL1\*C2\*AL3\*C4\*AL5\*C6\*AL7)/RL  
A(1)=1.000  
A(2)=AA6/AA7  
A(3)=AA5/AA7  
A(4)=AA4/AA7  
A(5)=AA3/AA7  
A(6)=AA2/AA7  
A(7)=AA1/AA7  
A(8)=AA0/AA7  
COEF1=A(8)

41 WRITE(6,41) 'THE COEF. OF H(S)=K/(S^7+A1\*S^6+A2\*S^5+A3\*S^4+A4\*S^3+A5\*S^2+A6\*S+A7) ARE',/  
40 WRITE(6,40) A(2),A(3),A(4),A(5),A(6),A(7),A(8)  
40 FORMAT(5X,'A1=',E12.5,3X,'A2=',E12.5,3X,'A3=',E12.5,3X,'A4=',E12.5,3X,'A5=',E12.5,3X,'A6=',E12.5,3X,'A7=',E12.5,/  
WRITE(6,43) COEF1

CALCULATION OF THE COEF. OF THE EQN.

$$H(S)=K/(S^2+A_1*S+B_1)(S^2+A_2*S+B_2)(S^2+A_3*S+B_3)(S+B_4)$$

43 FORMAT(5X,'K=',E12.5,/  
NDEG=7

SUBROUTINE ZPOLR GIVES THE ROOTS OF THE POLYNOMIAL OF DEGREE 'N'

CALL ZPOLR(A,NDEG,Y,IER)  
P1=-2.000\*REAL(Y(1))  
F1=REAL(Y(1))\*2+AI\*IMAG(Y(1))\*2  
P2=-2.000\*REAL(Y(3))  
F2=REAL(Y(3))\*2+AI\*IMAG(Y(3))\*2  
P3=-2.000\*REAL(Y(5))  
F3=REAL(Y(5))\*2+AI\*IMAG(Y(5))\*2  
P4=-REAL(Y(7))  
PP=1.000/AA7  
42 WRITE(6,42)  
42 FORMAT(5X,'THE COEF. OF H(S) ARE',/  
1 H(S)=K/((S^2+A1\*S+B1)(S^2+A2\*S+B2)(S^2+A3\*S+B3)(S+B4)',/  
WRITE(6,60) P1,F1,P2,F2,P3,F3,P4,COEF1

CALCULATION OF THE COEF OF H(Z) WITH SAMPLING PERIOD OF T

$$H(Z)=K*(1+Z^{-1})/(1+A_1*Z^{-1})(1+A_2*Z^{-1}+B_2*Z^{-2})(1+A_3*Z^{-1}+B_3*Z^{-2})(1+A_4*Z^{-1}+B_4*Z^{-2})$$



```

C
60  FORMAT(1X,'A1=',E12.5,1X,'B1=',E12.5,2X,'A2=',E12.5,1X,'B2=',E12.5
1,2X,'A3=',E12.5,1X,'B3=',E12.5,2X,'A4=',E12.5,3X,'K=',E12.5,/)
F=2.000/T
FF=F*F
B1=2.000*(F1-FF)/(FF+P1*F+F1)
D1=(FF-P1*F+F1)/(FF+P1*F+F1)
B2=2.000*(F2-FF)/(FF+P2*F+F2)
D2=(FF-P2*F+F2)/(FF+P2*F+F2)
B3=2.000*(F3-FF)/(FF+P3*F+F3)
D3=(FF-P3*F+F3)/(FF+P3*F+F3)
B4=(P4-F)/(P4+F)
COEF2=COEF1/((P4+F)*(FF+F*P1+F1)*(FF+F*P2+F2)*(FF+F*P3+F3))
WRITE(6,66)
66  FORMAT(5X,'THE COEF. OF Z**-1 AND Z**-2 IN QUADRATURE FORM AND TOT
IAL MULTIPLYING FACTOR K ARE...',/)
WRITE(6,60) B1,D1,B2,D2,B3,D3,B4,COEF2
C
UPP=UP*COEF2
C
C  FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
C  FREQUENCY INCREMENT
DLW=OMGAC/50
C
C  INITIAL VALUES IN THE FREQUENCY DOMAIN
W=0.000
C
C  ITERATION IN THE FREQUENCY DOMAIN
-----
DO 110 J=1,98
C
C  INITIAL VALUES IN TIME DOMAIN
UP1=UPP
TT=0.000
H=DCMPLX(0.000,0.000)
X1=0.0
X2=0.0
X3=0.0
X4=0.0
X5=0.0
X6=0.0
X7=0.0
C
C  ITERATION IN THE TIME DOMAIN
-----
DO 100 I=1,500
V1=UP1+X1
V2=V1+X2
V3=V2+X3
V=V3+X4
X1=UP1-B4*V1
X2=2*V1-B1*V2+X5
X3=2*V2-B2*V3+X6
X4=2*V3-B3*V+X7
X5=V1-D1*V2
X6=V2-D2*V3
X7=V3-D3*V
UP1=0.000
W1=W*TT
WT=DCMPLX(0.000,-W1)
Z=CDEXP(WT)
H=H+V*Z
TT=TT+T
C
100 CONTINUE
DA=CDABS(H)
C
C  ARRANGING THE OUTPUT DATA
DATA(J,1)=W
DATA(J,2)=DA
DATA(J,3)=0.000
C
C  ARRANGING THE DATA FOR PLOT SUBROUTINE

```





```

      DT(J,1)=DATA(J,1)
      DT(J,2)=DATA(J,2)
      DT(J,3)=DATA(J,3)
C
      W=W+DLW
110  CONTINUE
C
      WRITING AND PLOTTING THE FREQUENCY RESPONSE
      WRITE(6,54)
      FORMAT('1')
      WRITE(6,55)
      FORMAT(20X,'FREQ',21X,'OUTPUT NO 1',10X,'OUTPUT NO 2')
      WRITE(6,20) ((DATA(N,M),M=1,3),N=1,98)
      CALL GRAPHX(DT,98,4HFPEQ,4HMAGN)
      FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
20
C
      RETURN
      END

```



# FILERR

SUBROUTINE FILERR(UP,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA,NO  
\*BTS,ER,T)

SUBROUTINE FILERR TAKES THE ROOT MEAN SQUARE VALUE OF THE ERROR  
BETWEEN UNTRUNCATED AND TRUNCATED VALUE OF THE OUT PUT IN THE  
FREQUENCY DOMAIN AND ALSO PLOTS THE FINITE PRECESSION FREQUENCY  
RESPONCE

IMPLICIT REAL\*8 (A-H,O-Z)  
DIMENSION DATA2(210,3)  
DIMENSION DATA(210,3)

WRITE(6,54)  
FORMAT('1')  
WRITE(6,21)  
FORMAT(10X,'THE UNTRUNCATED VALUES ARE',/)  
WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7  
FORMAT(5X,'L1=',E12.5,3X,'C2=',E12.5,3X,'L3=',E12.5,3X,'C4=',E12.5  
\*,3X,'L5=',E12.5,3X,'C6=',E12.5,3X,'L7=',E12.5,/)   
WRITE(6,28) RS,RL  
FORMAT(20X,'RS=',E12.5,5X,'RL=',E12.5,/)

A=RS  
B=AL1  
C=C2  
D=AL3  
E=C4  
F=AL5  
G=C6  
H=AL7  
P=RL  
WRITE(6,33) NOBTS  
FORMAT(10X,' NO OF BITS =',I2,/)

PERFORMING THE TRUNCATION PROCESS TO REQUIRED NO. OF BITS ,  
'NOBTS'  
RS=TRUNC(RS,NOBTS)  
AL1=TRUNC(AL1,NOBTS)  
C2=TRUNC(C2,NOBTS)  
AL3=TRUNC(AL3,NOBTS)  
C4=TRUNC(C4,NOBTS)  
AL5=TRUNC(AL5,NOBTS)  
C6=TRUNC(C6,NOBTS)  
AL7=TRUNC(AL7,NOBTS)  
RL=TRUNC(RL,NOBTS)  
WRITE(6,25)  
FORMAT(10X,'THE TRUNCATED VALUES ARE')  
WRITE(6,22) AL1,C2,AL3,C4,AL5,C6,AL7  
WRITE(6,28) RS,RL

CALCULATION OF THE FREQUENCY RESPONCE WITH COMPLONENTS OF  
FINITE PRECESSION  
CALL FILTER(UP,RS,AL1,C2,AL3,C4,AL5,C6,AL7,RL,OMGAC,DATA2,T)

CALCULATION OF ROOT MEAN SQUARE ERROR  
ER=0.000  
DO 213 KK=1,98  
ER=ER+(DATA(KK,2)-DATA2(KK,2))\*2  
CONTINUE  
ER=ER/98.000  
ER=DSQRT(ER)

RS=A  
AL1=B  
C2=C  
AL3=D  
C4=E  
AL5=F  
C6=G  
AL7=H  
RL=P  
  
RETURN  
  
END



# APPENDIX 3

- A. Computer Program No. 10. Program to calculate the sensitivity function of the seventh order low pass wave digital filter with respect to both wave digital filter multiplier coefficients and the original filter component values.

```

C
C ***** FREQUENCY DOMAIN *****
C
C ***** SENSITIVITY FUNCTION OF THE FILTER *****
C
C WITH RESPECT TO L'S AND C'S ,RS,AND RL
C AND ALSO WITH RESPECT TO WAVE DIGITAL PARAMETERS
C ON THE SAME GRAPH FOR COMPARISON PURPOSES
C
C IMPLICIT REAL*8 (A-H,O-Z)
C
C DIMENSION DATA0(210,3)
C DIMENSION DATA1(210,3)
C DIMENSION DATA2(210,3)
C DIMENSION DATA3(210,3)
C DIMENSION DATA4(210,3)
C DIMENSION DATA5(210,3)
C DIMENSION DATA6(210,3)
C DIMENSION DATA7(210,3)
C DIMENSION DATA8(210,3)
C
C COMPLEX*16 H1,H2,H3,H4,H5,H6,H7,H8
C COMPLEX*16 T1,T2,T3,T4,T5,T6,T7,T8,T0
C COMPLEX*16 W1,Z,H
C
C ***** .5DB CHEBECHEV LOW-PASS FILTER WITH RS=1.0 *****
C
C RS=1.000
C AL1=1.789600
C C2=1.296100
C AL3=2.717700
C C4=1.384800
C AL5=2.717700
C C6=1.296100
C AL7=1.789600
C RL=1.000
C
C FILTER SCALE FACTOR FROM DATA IS COEF
C COEF=(RL+RS)/RL
C WRITE(6,37) COEF
37 FORMAT(7X,'THE FILTER SCALE FACTOR IS....',E12.5,/)
C
C WRITE(6,19) AL1,C2,AL3,C4,AL5,C6,AL7
19 FORMAT(5X,'SER L1=',F8.4,2X,'SHT C2=',F8.4,2X,'SER L3=',F8.4,2X,'
C *SHT C4=',F8.4,2X,'SER L5=',F8.4,2X,'SHT C6=',F8.4,2X,'SER L7=',F8.
C *4,/)
18 WRITE(6,18) RL,RS
C FORMAT(10X,'RL=',F8.4,5X,'RS=',F8.4,/)
C
C SPECIFY THE CUT OFF FREQUENCY IN RAD/SEC.
C OMGAC=1.000
C
C SAMPLING PERIOD IS DLTAT
C DLTAT=1.000
C
C SCALE TO NORMALIZE THE ELEMENT VALUES ,AS WELL AS TAKING INTO
C ACCOUNT THE EFFECT OF SAMPLING TIME AND FREQUENCY SCALE
C SCALE=1.000/DTAN(OMGAC*DLTAT/2)
C AL1=AL1*SCALE
C C2=C2*SCALE
C AL3=AL3*SCALE
C C4=C4*SCALE
C AL5=AL5*SCALE
C C6=C6*SCALE
C AL7=AL7*SCALE
C
C CALCULATION TO FIND THE TERMINATING RESISTANCE OF EACH ELEMENT
C AND WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS
C WITH NO DELAY FREE PATH ON PORT TWO

```

\* Subroutine Graphyx is given in Computer Program No. 1, Appendix 1.



```

R1=RS
R2=R1+AL1
SIGMA1=R1/R2
G2=1.000/R2
G3=G2+C2
SIGMA2=G2/G3
R3=1.000/G3
R4=R3+AL3
SIGMA3=R3/R4
G4=1.000/R4
G5=G4+C4
SIGMA4=G4/G5
R5=1.000/G5
R6=R5+AL5
SIGMA5=R5/R6
G6=1.000/R6
G7=G6+C6
SIGMA6=G6/G7
R7=1.000/G7
R8=R7+AL7
SIGMA7=R7/R8
PHI=(R1-R8)/(R1+R8)
WRITE(6,42)

```

```

42  FORMAT(5X,'THE WAVE DIGITAL FILTER MULTIPLIER COEFFICIENTS ARE....
*1,/)
10  WRITE(6,10) SIGMA1,SIGMA2,SIGMA3,SIGMA4,SIGMA5,SIGMA6,SIGMA7,PHI
    FORMAT(/,4X,'SIGMA1=',F6.4,4X,'SIGMA2=',F6.4,4X,'SIGMA3=',F6.4,4X,
*1'SIGMA4=',F6.4,4X,'SIGMA5=',F6.4,4X,'SIGMA6=',F6.4,4X,'SIGMA7=',F6
*.4,4X,'PHI=',F6.4)

```

```

C
C
C  IMPULSE INPUT TO THE FILTER WITH MAGNITUDE OF AS=1
AS=1.000
UP=AS*COEF

```

```

C
C  FREQUENCY RANGE IS CHOSEN TO BE TWICE THE CRITICAL FREQUENCY
FREQUENCY INCREMENT
DLTAW=OMGAC/50.00

```

```

C
C
C  PREMULIPLICATION OF
COEFFICIENTS OF PARTIAL DIFFERENTIATION OF SIGMA'S AND PHI
WITH RESPECT TO L'S AND C'S ,RS,AND RL

```

```

A1=AL1/(R1+AL1)**2
A2=A1*C2*R1/(SIGMA1+R1*C2)**2
A3=A2*R2*AL3/(SIGMA2*R2+AL3)**2
A4=A3*G3*C4/(SIGMA3*G3+C4)**2
A5=A4*R4*AL5/(SIGMA4*R4+AL5)**2
A6=A5*G5*C6/(SIGMA5*G5+C6)**2
A7=A6*R6*AL7/(SIGMA6*R6+AL7)**2
A8=-2*A7*G7*RL/(SIGMA7*G7+RL)**2

C
B1=-R1/(R1+AL1)**2
B2=-R1*R1*C2/((SIGMA1+R1*C2)*(R1+AL1))**2
B3=B2*R2*AL3/(SIGMA2*R2+AL3)**2
B4=B3*G3*C4/(SIGMA3*G3+C4)**2
B5=B4*R4*AL5/(SIGMA4*R4+AL5)**2
B6=B5*G5*C6/(SIGMA5*G5+C6)**2
B7=B6*R6*AL7/(SIGMA6*R6+AL7)**2
B8=-2*B7*G7*RL/(SIGMA7*G7+RL)**2

C
D2=-G2/(G2+C2)**2
D3=D2*R2*AL3/(SIGMA2*R2+AL3)**2
D4=D3*G3*C4/(SIGMA3*G3+C4)**2
D5=D4*R4*AL5/(SIGMA4*R4+AL5)**2
D6=D5*G5*C6/(SIGMA5*G5+C6)**2
D7=D6*R6*AL7/(SIGMA6*R6+AL7)**2
D8=-2*D7*G7*RL/(SIGMA7*G7+RL)**2

C
E3=-R3/(R3+AL3)**2
E4=E3*G3*C4/(SIGMA3*G3+C4)**2

```





```

E5=E4*R4*AL5/(SIGMA4*R4+AL5)**2
E6=E5*G5*C6/(SIGMA5*G5+C6)**2
E7=E6*R6*AL7/(SIGMA6*R6+AL7)**2
E8=-2*E7*G7*RL/(SIGMA7*G7+RL)**2
C
P4=-G4/(G4+C4)**2
P5=P4*R4*AL5/(SIGMA4*R4+AL5)**2
P6=P5*G5*C6/(SIGMA5*G5+C6)**2
P7=P6*R6*AL7/(SIGMA6*R6+AL7)**2
P8=-2*P7*G7*RL/(SIGMA7*G7+RL)**2
C
Q5=-R5/(R5+AL5)**2
Q6=Q5*G5*C6/(SIGMA5*G5+C6)**2
Q7=Q6*R6*AL7/(SIGMA6*R6+AL7)**2
Q8=-2*Q7*G7*RL/(SIGMA7*G7+RL)**2
C
S6=-G6/(G6+C6)**2
S7=S6*R6*AL7/(SIGMA6*R6+AL7)**2
S8=-2*S7*G7*RL/(SIGMA7*G7+RL)**2
C
U7=-R7/(R7+AL7)**2
U8=-2*U7*G7*RL/(SIGMA7*G7+RL)**2
C
V8=2*G7*SIGMA7/(RL+G7*SIGMA7)**2
C
INITIAL VALUES IN FREQUENCY DOMAIN
-----
W=0.000
C
ITERATION IN THE FREQUENCY DOMAIN
*****
DO 110 J=1,98
C
INITIAL VALUES IN THE TIME DOMAIN
-----
C11=UP
C
TT=0.000
H=DCMPLX(0.000,0.000)
H1=DCMPLX(0.000,0.000)
H2=DCMPLX(0.000,0.000)
H3=DCMPLX(0.000,0.000)
H4=DCMPLX(0.000,0.000)
H5=DCMPLX(0.000,0.000)
H6=DCMPLX(0.000,0.000)
H7=DCMPLX(0.000,0.000)
H8=DCMPLX(0.000,0.000)
C
Y11=0.000
Y13=0.000
Y14=0.000
Y23=0.000
Y24=0.000
Y33=0.000
Y34=0.000
Y43=0.000
Y44=0.000
Y53=0.000
Y54=0.000
Y63=0.000
Y64=0.000
Y72=0.000
Y73=0.000
Y74=0.000
C
DX111=0.000
DX113=0.000
DX114=0.000
DX123=0.000
DX124=0.000
DX133=0.000
DX134=0.000

```



DX143=0.000  
 DX144=0.000  
 DX153=0.000  
 DX154=0.000  
 DX163=0.000  
 DX164=0.000  
 DX172=0.000  
 DX173=0.000  
 DX174=C.000

C  
 DX211=0.000  
 DX213=0.000  
 DX214=C.000  
 DX223=0.000  
 DX224=C.000  
 DX233=0.000  
 DX234=0.000  
 DX243=C.000  
 DX244=0.000  
 DX253=C.000  
 DX254=C.000  
 DX263=0.000  
 DX264=0.000  
 DX272=0.000  
 DX273=0.000  
 DX274=0.000

C  
 DX311=0.000  
 DX313=0.000  
 DX314=0.000  
 DX323=0.000  
 DX324=0.000  
 DX333=0.000  
 DX334=C.000  
 DX343=0.000  
 DX344=0.000  
 DX353=0.000  
 DX354=0.000  
 DX363=0.000  
 DX364=0.000  
 DX372=0.000  
 DX373=0.000  
 DX374=0.000

C  
 DX411=0.000  
 DX413=0.000  
 DX414=C.000  
 DX423=0.000  
 DX424=0.000  
 DX433=0.000  
 DX434=0.000  
 DX443=0.000  
 DX444=0.000  
 DX453=0.000  
 DX454=0.000  
 DX463=0.000  
 DX464=0.000  
 DX472=0.000  
 DX473=0.000  
 DX474=0.000

C  
 DX511=0.000  
 DX513=0.000  
 DX514=0.000  
 DX523=0.000  
 DX524=0.000  
 DX533=C.000  
 DX534=0.000  
 DX543=0.000  
 DX544=0.000  
 DX553=0.000  
 DX554=C.000  
 DX563=0.000



```

DX 564=0.000
DX 572=0.000
DX 573=0.000
DX 574=0.000

```

C

```

DX 611=0.000
DX 613=0.000
DX 614=0.000
DX 623=0.000
DX 624=0.000
DX 633=0.000
DX 634=0.000
DX 643=0.000
DX 644=0.000
DX 653=0.000
DX 654=0.000
DX 663=0.000
DX 664=0.000
DX 672=0.000
DX 673=0.000
DX 674=0.000

```

C

```

DX 711=0.000
DX 713=0.000
DX 714=0.000
DX 723=0.000
DX 724=0.000
DX 733=0.000
DX 734=0.000
DX 743=0.000
DX 744=0.000
DX 753=0.000
DX 754=0.000
DX 763=0.000
DX 764=0.000
DX 772=0.000
DX 773=0.000
DX 774=0.000

```

C

```

DX 811=0.000
DX 813=0.000
DX 814=0.000
DX 823=0.000
DX 824=0.000
DX 833=0.000
DX 834=0.000
DX 843=0.000
DX 844=0.000
DX 853=0.000
DX 854=0.000
DX 863=0.000
DX 864=0.000
DX 872=0.000
DX 873=0.000
DX 874=0.000

```

C

C

C

C

```

ITERATION IN THE TIME DOMAIN
*****
DO 10C I=1,480

```

```

D12=C11+Y11-Y23+SIGMA1*(Y23-Y14)
D22=Y33+SIGMA2*(D12+Y14-Y33-Y24)
D32=D22+Y24-Y43+SIGMA3*(Y43-Y34)
D42=Y53+SIGMA4*(D32+Y34-Y53-Y44)
D52=D42+Y44-Y63+SIGMA5*(Y63-Y54)
D62=Y73+SIGMA6*(D52+Y54-Y73-Y64)
D72=D62+Y64-Y72+SIGMA7*(Y72-Y74)
C72=D72*PHI
D71=D62+SIGMA7*(C72-D62+Y72-Y73)
D61=D71-D52+Y73+SIGMA6*(D52-Y63)
D51=D42+SIGMA5*(D61-D42+Y63-Y53)
D41=D51-D32+Y53+SIGMA4*(D32-Y43)

```



$D31 = D22 + \text{SIGMA}3 * (D41 - D22 + Y43 - Y33)$   
 $D21 = D31 - D12 + Y33 + \text{SIGMA}2 * (D12 - Y23)$   
 $D11 = C11 + \text{SIGMA}1 * (D21 - C11 + Y23 - Y13)$

#### DIFFERENTIATION WITH RESPECT TO SIGMA1

$DB112 = Y23 - Y14 - DX123 + \text{SIGMA}1 * (DX123 - DX114)$   
 $DB122 = DX133 + \text{SIGMA}2 * (DB112 + DX114 - DX123 - DX124)$   
 $DB132 = DB122 + DX124 - DX143 + \text{SIGMA}3 * (DX143 - DX134)$   
 $DB142 = DX153 + \text{SIGMA}4 * (DB132 + DX134 - DX153 - DX144)$   
 $DB152 = DB142 + DX144 - DX163 + \text{SIGMA}5 * (DX163 - DX154)$   
 $DB162 = DX173 + \text{SIGMA}6 * (DB152 + DX154 - DX173 - DX164)$   
 $DB172 = DB162 + DX164 - DX172 + \text{SIGMA}7 * (DX172 - DX174)$   
 $DA172 = DB172 * \text{PHI}$   
 $DB171 = DB162 + \text{SIGMA}7 * (DA172 - DB162 + DX172 - DX173)$   
 $DB161 = DB171 - DB152 + DX173 + \text{SIGMA}6 * (DB152 - DX163)$   
 $DB151 = DB142 + \text{SIGMA}5 * (DB161 - DB142 + DX163 - DX153)$   
 $DB141 = DB151 - DB132 + DX153 + \text{SIGMA}4 * (DB132 - DX143)$   
 $DB131 = DB122 + \text{SIGMA}3 * (DB141 - DB122 + DX143 - DX133)$   
 $DB121 = DB131 - DB112 + DX133 + \text{SIGMA}2 * (DB112 - DX123)$   
 $DB111 = D21 - C11 + Y23 - Y13 + \text{SIGMA}1 * (DB121 + DX123 - DX113)$

#### DIFFERENTIATION WITH RESPECT TO SIGMA2

$DB212 = -DX223 + \text{SIGMA}1 * (DX223 - DX214)$   
 $DB222 = DX233 + \text{SIGMA}2 * (DB212 + DX214 - DX233 - DX224) + D12 + Y14 - Y33 - Y24$   
 $DB232 = DB222 + DX224 - DX243 + \text{SIGMA}3 * (DX243 - DX234)$   
 $DB242 = DX253 + \text{SIGMA}4 * (DB232 + DX234 - DX253 - DX244)$   
 $DB252 = DB242 + DX244 - DX263 + \text{SIGMA}5 * (DX263 - DX254)$   
 $DB262 = DX273 + \text{SIGMA}6 * (DB252 + DX254 - DX273 - DX264)$   
 $DB272 = DB262 + DX264 - DX272 + \text{SIGMA}7 * (DX272 - DX274)$   
 $DA272 = DB272 * \text{PHI}$   
 $DB271 = DB262 + \text{SIGMA}7 * (DA272 - DB262 + DX272 - DX273)$   
 $DB261 = DB271 - DB252 + DX273 + \text{SIGMA}6 * (DB252 - DX263)$   
 $DB251 = DB242 + \text{SIGMA}5 * (DB261 - DB242 + DX263 - DX253)$   
 $DB241 = DB251 - DB232 + DX253 + \text{SIGMA}4 * (DB232 - DX243)$   
 $DB231 = DB222 + \text{SIGMA}3 * (DB241 - DB222 + DX243 - DX233)$   
 $DB221 = DB231 - DB212 + DX233 + \text{SIGMA}2 * (DB212 - DX223) + D12 - Y23$   
 $DB211 = \text{SIGMA}1 * (DB221 + DX223 - DX213)$

#### DIFFERENTIATION WITH RESPECT TO SIGMA3

$DB312 = -DX323 + \text{SIGMA}1 * (DX323 - DX314)$   
 $DB322 = DX333 + \text{SIGMA}2 * (DB312 + DX314 - DX333 - DX324)$   
 $DB332 = DB322 + DX324 - DX343 + \text{SIGMA}3 * (DX343 - DX334) + Y43 - Y34$   
 $DB342 = DX353 + \text{SIGMA}4 * (DB332 + DX334 - DX353 - DX344)$   
 $DB352 = DB342 + DX344 - DX363 + \text{SIGMA}5 * (DX363 - DX354)$   
 $DB362 = DX373 + \text{SIGMA}6 * (DB352 + DX354 - DX373 - DX364)$   
 $DB372 = DB362 + DX364 - DX372 + \text{SIGMA}7 * (DX372 - DX374)$   
 $DA372 = DB372 * \text{PHI}$   
 $DB371 = DB362 + \text{SIGMA}7 * (DA372 - DB362 + DX372 - DX373)$   
 $DB361 = DB371 - DB352 + DX373 + \text{SIGMA}6 * (DB352 - DX363)$   
 $DB351 = DB342 + \text{SIGMA}5 * (DB361 - DB342 + DX363 - DX353)$   
 $DB341 = DB351 - DB332 + DX353 + \text{SIGMA}4 * (DB332 - DX343)$   
 $DB331 = DB322 + \text{SIGMA}3 * (DB341 - DB322 + DX343 - DX333) + D41 - D22 + Y43 - Y33$   
 $DB321 = DB331 - DB312 + DX333 + \text{SIGMA}2 * (DB312 - DX323)$   
 $DB311 = \text{SIGMA}1 * (DB321 + DX323 - DX313)$

#### DIFFERENTIATION WITH RESPECT TO SIGMA4

$DB412 = -DX423 + \text{SIGMA}1 * (DX423 - DX414)$   
 $DB422 = DX433 + \text{SIGMA}2 * (DB412 + DX414 - DX433 - DX424)$   
 $DB432 = DB422 + DX424 - DX443 + \text{SIGMA}3 * (DX443 - DX434)$   
 $DB442 = DX453 + \text{SIGMA}4 * (DB432 + DX434 - DX453 - DX444) + D32 + Y34 - Y53 - Y44$   
 $DB452 = DB442 + DX444 - DX463 + \text{SIGMA}5 * (DX463 - DX454)$   
 $DB462 = DX473 + \text{SIGMA}6 * (DB452 + DX454 - DX473 - DX464)$   
 $DB472 = DB462 + DX464 - DX472 + \text{SIGMA}7 * (DX472 - DX474)$   
 $DA472 = DB472 * \text{PHI}$





$DB471 = DB462 + SIGMA7 * (DA472 - DB462 + DX472 - DX473)$   
 $DB461 = DB471 - DB452 + DX473 + SIGMA6 * (DB452 - DX463)$   
 $DB451 = DB442 + SIGMA5 * (DB461 - DB442 + DX463 - DX453)$   
 $DB441 = DB451 - DB432 + DX453 + SIGMA4 * (DB432 - DX443) + D32 - Y43$   
 $DB431 = DB422 + SIGMA3 * (DB441 - DB422 + DX443 - DX433)$   
 $DB421 = DB431 - DB412 + DX433 + SIGMA2 * (DB412 - DX423)$   
 $DB411 = SIGMA1 * (DB421 + DX423 - DX413)$

#### DIFFERENTIATION WITH RESPECT TO SIGMA5

$DB512 = -DX523 + SIGMA1 * (DX523 - DX514)$   
 $DB522 = DX533 + SIGMA2 * (DB512 + DX514 - DX533 - DX524)$   
 $DB532 = DB522 + DX524 - DX543 + SIGMA3 * (DX543 - DX534)$   
 $DB542 = DX553 + SIGMA4 * (DB532 + DX534 - DX553 - DX544)$   
 $DB552 = DB542 + DX544 - DX563 + SIGMA5 * (DX563 - DX554) + Y63 - Y54$   
 $DB562 = DX573 + SIGMA6 * (DB552 + DX554 - DX573 - DX564)$   
 $DB572 = DB562 + DX564 - DX572 + SIGMA7 * (DX572 - DX574)$   
 $DA572 = DB572 * PHI$   
 $DB571 = DB562 + SIGMA7 * (DA572 - DB562 + DX572 - DX573)$   
 $DB561 = DB571 - DB552 + DX573 + SIGMA6 * (DB552 - DX563)$   
 $DB551 = DB542 + SIGMA5 * (DB561 - DB542 + DX563 - DX553) + D61 - D42 + Y63 - Y53$   
 $DB541 = DB551 - DB532 + DX553 + SIGMA4 * (DB532 - DX543)$   
 $DB531 = DB522 + SIGMA3 * (DB541 - DB522 + DX543 - DX533)$   
 $DB521 = DB531 - DB512 + DX533 + SIGMA2 * (DB512 - DX523)$   
 $DB511 = SIGMA1 * (DB521 + DX523 - DX513)$

#### DIFFERENTIATION WITH RESPECT TO SIGMA6

$DB612 = -DX623 + SIGMA1 * (DX623 - DX614)$   
 $DB622 = DX633 + SIGMA2 * (DB612 + DX614 - DX633 - DX624)$   
 $DB632 = DB622 + DX624 - DX643 + SIGMA3 * (DX643 - DX634)$   
 $DB642 = DX653 + SIGMA4 * (DB632 + DX634 - DX653 - DX644)$   
 $DB652 = DB642 + DX644 - DX663 + SIGMA5 * (DX663 - DX654)$   
 $DB662 = DX673 + SIGMA6 * (DB652 + DX654 - DX673 - DX664) + D52 + Y54 - Y73 - Y64$   
 $DB672 = DB662 + DX664 - DX672 + SIGMA7 * (DX672 - DX674)$   
 $DA672 = DB672 * PHI$   
 $DB671 = DB662 + SIGMA7 * (DA672 - DB662 + DX672 - DX673)$   
 $DB661 = DB671 - DB652 + DX673 + SIGMA6 * (DB652 - DX663) + D52 - Y63$   
 $DB651 = DB642 + SIGMA5 * (DB661 - DB642 + DX663 - DX653)$   
 $DB641 = DB651 - DB632 + DX653 + SIGMA4 * (DB632 - DX643)$   
 $DB631 = DB622 + SIGMA3 * (DB641 - DB622 + DX643 - DX633)$   
 $DB621 = DB631 - DB612 + DX633 + SIGMA2 * (DB612 - DX623)$   
 $DB611 = SIGMA1 * (DB621 + DX623 - DX613)$

#### DIFFERENTIATION WITH RESPECT TO SIGMA7

$DB712 = -DX723 + SIGMA1 * (DX723 - DX714)$   
 $DB722 = DX733 + SIGMA2 * (DB712 + DX714 - DX733 - DX724)$   
 $DB732 = DB722 + DX724 - DX743 + SIGMA3 * (DX743 - DX734)$   
 $DB742 = DX753 + SIGMA4 * (DB732 + DX734 - DX753 - DX744)$   
 $DB752 = DB742 + DX744 - DX763 + SIGMA5 * (DX763 - DX754)$   
 $DB762 = DX773 + SIGMA6 * (DB752 + DX754 - DX773 - DX764)$   
 $DB772 = DB762 + DX764 - DX772 + SIGMA7 * (DX772 - DX774) + Y72 - Y74$   
 $DA772 = DB772 * PHI$   
 $DB771 = DB762 + SIGMA7 * (DA772 - DB762 + DX772 - DX773) + C72 - C62 + Y72 - Y73$   
 $DB761 = DB771 - DB752 + DX773 + SIGMA6 * (DB752 - DX763)$   
 $DB751 = DB742 + SIGMA5 * (DB761 - DB742 + DX763 - DX753)$   
 $DB741 = DB751 - DB732 + DX753 + SIGMA4 * (DB732 - DX743)$   
 $DB731 = DB722 + SIGMA3 * (DB741 - DB722 + DX743 - DX733)$   
 $DB721 = DB731 - DB712 + DX733 + SIGMA2 * (DB712 - DX723)$   
 $DB711 = SIGMA1 * (DB721 + DX723 - DX713)$

#### DIFFERENTIATION WITH RESPECT TO PHI

$DB812 = -DX823 + SIGMA1 * (DX823 - DX814)$   
 $DB822 = DX833 + SIGMA2 * (DB812 + DX814 - DX833 - DX824)$   
 $DB832 = DB822 + DX824 - DX843 + SIGMA3 * (DX843 - DX834)$   
 $DB842 = DX853 + SIGMA4 * (DB832 + DX834 - DX853 - DX844)$   
 $DB852 = DB842 + DX844 - DX863 + SIGMA5 * (DX863 - DX854)$



```

D8862=D8873+SIGMA6*(D8852+DX854-DX873-DX864)
D8872=D8862+DX864-DX872+SIGMA7*(DX872-DX874)
DA872=D8872*PHI+D72
D8871=D8862+SIGMA17*(DA872-D8862+DX872-DX873)
D8861=D8871-D8852+DX873+SIGMA6*(D8852-DX863)
D8851=D8842+SIGMA15*(D8861-D8842+DX863-DX853)
D8841=D8851-D8832+DX853+SIGMA14*(D8832-DX843)
D8831=D8822+SIGMA13*(D8841-D8822+DX843-DX833)
D8821=D8831-D8812+DX833+SIGMA2*(D8812-DX823)
D8811=SIGMA1*(D8821+DX823-DX813)

```

UPDATED VALUES FOR NEXT ITERATION

```

Y11=C11
Y13=D11
Y14=D12
Y23=D21
Y24=D22
Y33=D31
Y34=D32
Y43=D41
Y44=D42
Y53=D51
Y54=D52
Y63=D61
Y64=D62
Y72=C72
Y73=D71
Y74=D72

```

```

DX113=D8111
DX114=D8112
DX123=D8121
DX124=D8122
DX133=D8131
DX134=D8132
DX143=D8141
DX144=D8142
DX153=D8151
DX154=D8152
DX163=D8161
DX164=D8162
DX172=D8172
DX173=D8171
DX174=D8172

```

```

DX213=D8211
DX214=D8212
DX223=D8221
DX224=D8222
DX233=D8231
DX234=D8232
DX243=D8241
DX244=D8242
DX253=D8251
DX254=D8252
DX263=D8261
DX264=D8262
DX272=D8272
DX273=D8271
DX274=D8272

```

```

DX313=D8311
DX314=D8312
DX323=D8321
DX324=D8322
DX333=D8331
DX334=D8332
DX343=D8341
DX344=D8342
DX353=D8351
DX354=D8352

```



DX363=08361  
DX364=08362  
DX372=0A372  
DX373=08371  
DX374=08372

C  
DX413=08411  
DX414=08412  
DX423=08421  
DX424=08422  
DX433=08431  
DX434=08432  
DX443=08441  
DX444=08442  
DX453=08451  
DX454=08452  
DX463=08461  
DX464=08462  
DX472=0A472  
DX473=08471  
DX474=08472

C  
DX513=08511  
DX514=08512  
DX523=08521  
DX524=08522  
DX533=08531  
DX534=08532  
DX543=08541  
DX544=08542  
DX553=08551  
DX554=08552  
DX563=08561  
DX564=08562  
DX572=0A572  
DX573=08571  
DX574=08572

C  
DX613=08611  
DX614=08612  
DX623=08621  
DX624=08622  
DX633=08631  
DX634=08632  
DX643=08641  
DX644=08642  
DX653=08651  
DX654=08652  
DX663=08661  
DX664=08662  
DX672=0A672  
DX673=08671  
DX674=08672  
DX713=08711  
DX714=08712  
DX723=08721  
DX724=08722  
DX733=08731  
DX734=08732  
DX743=08741  
DX744=08742  
DX753=08751  
DX754=08752  
DX763=08761  
DX764=08762  
DX772=0A772  
DX773=08771  
DX774=08772

C  
DX813=08811  
DX814=08812  
DX823=08821  
DX824=08822



```

DX833=08831
DX834=08832
DX843=08841
DX844=08842
DX853=08851
DX854=08852
DX863=08861
DX864=08862
DX872=08872
DX873=08871
DX874=08872

C
C C11=0.000
WT=W*TT
W1=DC*PLX(0.000,-WT)
Z=CDEXP(W1)
H1=H1+08172*Z
H2=H2+08272*Z
H3=H3+08372*Z
H4=H4+08472*Z
H5=H5+08572*Z
H6=H6+08672*Z
H7=H7+08772*Z
H8=H8+08872*Z
H=H+072*Z
C UPDATING THE TIME INCREMENT
TT=TT+DLTAT
C
C 100 CONTINUE
H1=H1*(1.000+PHI)/2
H2=H2*(1.000+PHI)/2
H3=H3*(1.000+PHI)/2
H4=H4*(1.000+PHI)/2
H5=H5*(1.000+PHI)/2
H6=H6*(1.000+PHI)/2
H7=H7*(1.000+PHI)/2
H8=H8*(1.000+PHI)/2+H/2
C
C C
C C ARRANGING THE OUTPUT DATA IN FREQUENCY DOMAIN
C C -----
DATA0(J,1)=W
DATA1(J,1)=W
DATA2(J,1)=W
DATA3(J,1)=W
DATA4(J,1)=W
DATA5(J,1)=W
DATA6(J,1)=W
DATA7(J,1)=W
DATA8(J,1)=W
C
C C
C C ARRANGING THE NORMALIZED OUTPUT DERIVATIVES WITH RESPECT TO
C C SIGMA'S, AND PHI
DATA0(J,2)=C.000
DATA1(J,2)=CDABS(H1)*SIGMA1
IF(RS.LE.1.00-15) DATA1(J,2)=CDABS(H1)
C IF RS=0 THEN SIGMA1=0 THUS FOR THIS CASE ONLY NON NORMALIZED
C VALUES FOR DERIVATIVES ARE CALCULATED AND PLOTTED
DATA2(J,2)=CDABS(H2)*SIGMA2
DATA3(J,2)=CDABS(H3)*SIGMA3
DATA4(J,2)=CDABS(H4)*SIGMA4
DATA5(J,2)=CDABS(H5)*SIGMA5
DATA6(J,2)=CDABS(H6)*SIGMA6
DATA7(J,2)=CDABS(H7)*SIGMA7
DATA8(J,2)=CDABS(H8)*DABS(PHI)
C
C C
T0=A1*H1+A2*H2+A3*H3+A4*H4+A5*H5+A6*H6+A7*H7+A8*H8
T1=B1*H1+B2*H2+B3*H3+B4*H4+B5*H5+B6*H6+B7*H7+B8*H8

```





```

T2=D2*H2+D3*H3+D4*H4+D5*H5+D6*H6+D7*H7+D8*H8
T3=E3*H3+E4*H4+E5*H5+E6*H6+E7*H7+E8*H8
T4=P4*H4+P5*H5+P6*H6+P7*H7+P8*H8
T5=Q5*H5+Q6*H6+Q7*H7+Q8*H8
T6=S6*H6+S7*H7+S8*H8
T7=U7*H7+U8*H8
T8=V8*H8

```

```

ARRANGING THE NORMALIZED OUTPUT DERIVATIVES WITH RESPECT TO
L'S,C'S,RS,AND RL
NOTE THAT FOR THE CASE OF RS=0 NON NORMALIZED VALUES ARE
ARRANGED FOR SECTION ONE ONLY

```

```

DATA0(J,3)=CDABS(T0)*RS
IF(RS.LE.1.00-15) DATA0(J,3)=CDABS(T0)
DATA1(J,3)=CDABS(T1)*AL1
IF(RS.LE.1.00-15) DATA1(J,3)=CDABS(T1)
DATA2(J,3)=CDABS(T2)*C2
DATA3(J,3)=CDABS(T3)*AL3
DATA4(J,3)=CDABS(T4)*C4
DATA5(J,3)=CDABS(T5)*AL5
DATA6(J,3)=CDABS(T6)*C6
DATA7(J,3)=CDABS(T7)*AL7
DATA8(J,3)=CDABS(T8)*RL

```

```

UPDATING THE FREQUENCY INCREMENT
W=W+DLTAW
CONTINUE

```

```

WRITE(6,54)
FORMAT('1')
IF(RS.GT.1.00-15) GO TO 555
WRITE(6,50)
FORMAT(35X,'NON NORMALIZED VALUES',//)
WRITE(6,49)
FORMAT(20X,'FREQ',40X,'D(O/P)/D(RS)',//)
GO TO 666
555 CONTINUE
WRITE(6,48)
FORMAT(20X,'FREQ',40X,'(D(O/P)/D(RS))*RS',//)
666 CONTINUE
WRITE(6,20) ((DATA0(N,M),M=1,3),N=1,98)
CALL GRAPHX(DAT0,98,4HFREQ,4HMAG.)

```

```

WRITE(6,54)
IF(RS.GT.1.00-15) GO TO 333
WRITE(6,50)
WRITE(6,51)
51 FORMAT(20X,'FREQ',15X,'D(O/P)/D(SIGMA1)',9X,'D(O/P)/D(RL)',//)
GO TO 444
333 CONTINUE
WRITE(6,55)
55 FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA1))*SIGMA1',4X,'(D(O/P)/D(L1
*) ) *L1',//)
444 CONTINUE
WRITE(6,20) ((DATA1(N,M),M=1,3),N=1,98)
CALL GRAPHX(DAT1,98,4HFREQ,4HMAG.)

```

```

WRITE(6,54)
WRITE(6,56)
56 FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA2))*SIGMA2',4X,'(D(O/P)/D(L2
*) ) *C2',//)
WRITE(6,20) ((DATA2(N,M),M=1,3),N=1,98)
CALL GRAPHX(DAT2,98,4HFREQ,4HMAG.)

```

```

WRITE(6,54)
WRITE(6,57)
57 FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA3))*SIGMA3',4X,'(D(O/P)/D(L3
*) ) *L3',//)
WRITE(6,20) ((DATA3(N,M),M=1,3),N=1,98)

```



```

C      CALL GRAPHX(DATA3,98,4HFREQ,4HMAG.)
C      WRITE(6,54)
C      WRITE(6,58)
58     FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA4))*SIGMA4',4X,'(D(O/P)/D(L4
*) ) *C4',//)
C      WRITE(6,20) ((DATA4(N,M),M=1,3),N=1,98)
C      CALL GRAPHX(DATA4,98,4HFREQ,4HMAG.)
C      WRITE(6,54)
C      WRITE(6,59)
59     FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA5))*SIGMA5',4X,'(D(C/P)/D(L5
*) ) *L5',//)
C      WRITE(6,20) ((DATA5(N,M),M=1,3),N=1,98)
C      CALL GRAPHX(DATA5,98,4HFREQ,4HMAG.)
C      WRITE(6,54)
C      WRITE(6,60)
60     FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA6))*SIGMA6',4X,'(D(O/P)/D(L6
*) ) *C6',//)
C      WRITE(6,20) ((DATA6(N,M),M=1,3),N=1,98)
C      CALL GRAPHX(DATA6,98,4HFREQ,4HMAG.)
C      WRITE(6,54)
C      WRITE(6,61)
61     FORMAT(20X,'FREQ',11X,'(D(O/P)/D(SIGMA7))*SIGMA7',4X,'(D(C/P)/D(L7
*) ) *L7',//)
C      WRITE(6,20) ((DATA7(N,M),M=1,3),N=1,98)
C      CALL GRAPHX(DATA7,98,4HFREQ,4HMAG.)
C      WRITE(6,54)
C      WRITE(6,62)
62     FORMAT(20X,'FREQ',11X,'(D(O/P)/D(PHI))*ABS(PHI)',4X,'(D(O/P)/D(RL
*) ) *RL',//)
C      WRITE(6,20) ((DATA8(N,M),M=1,3),N=1,98)
C      CALL GRAPHX(DATA8,98,4HFREQ,4HMAG.)
C
C 20     FORMAT(20X,E12.5,10X,E12.5,10X,E12.5)
C
C      STOP
C      END

```



# INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3. Professor Sydney R. Parker, Code 62Px Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	10
4. Professor Richard W. Hamming, Code 52Hg Department of Computer Sciences Naval Postgraduate School Monterey, California 93940	1
5. CDR Masoud Sanaie-Fard Iranian Navy c/o National Iranian Navy Training Office Suite 1110 800 North Quincy Street Arlington, Virginia 22203	3
6. Department of Electrical Engineering Univerity of Tehran Tehran, IRAN	1
7. Department of Electrical Engineering University of Azerbaijan Tabriz, IRAN	1
8. Professor B. Peikari Electrical Engineering Department Institute of Technology Southern Methodist University Dallas, Texas 75275	1
9. Professor Donald E. Kirk, Code 62 Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1



Thesis  
S1567 Sanaie-Fard 183810  
c.1 An in-depth study of  
the sensitivity of wave  
digital filters.

Thesis  
S1567 Sanaie-Fard 183810  
c.1 An in-depth study of  
the sensitivity of wave  
digital filters.



thesS1567  
An in-depth study of the sensitivity of



3 2768 001 00153 0  
DUDLEY KNOX LIBRARY